A High-Linearity Digital-to-Time Converter Technique: Constant-Slope Charging

Jiayoon Zhiyu Ru, *Member, IEEE*, Claudia Palattella, *Student Member, IEEE*, Paul Geraedts, *Member, IEEE*, Eric Klumperink, *Senior Member, IEEE*, and Bram Nauta, *Fellow, IEEE*

***Abstract*—A digital-to-time converter (DTC) controls time delay by a digital code, which is useful, for example, in a sampling oscilloscope, fractional-N PLL, or time-interleaved ADC. This paper proposes *constant-slope charging* as a method to realize a DTC with intrinsically better integral non-linearity (INL) compared to the popular variable-slope method. The proposed DTC chip realized in 65 nm CMOS consists of a voltage-controlled variable-delay element (DTC-core) driven by a 10 bit digital-to-analog converter. Measurements with a 55 MHz crystal clock demonstrate a full-scale delay programmable from 19 ps to 189 ps with a resolution from 19 fs to 185 fs. As available oscilloscopes are not good enough to reliably measure such high timing resolution, a frequency-domain method has been developed that modulates a DTC edge and derives INL from spur strength. An INL of 0.17% at 189 ps full-scale delay and 0.34% at 19 ps are measured, representing 8–9 bit effective INL-limited resolution. Output rms jitter is better than 210 fs limited by the test setup, while the DTC consumes 1.8 mW.**

***Index Terms*—Constant slope, delay, delay measurement, digital-to-time converter, DTC, INL, integral nonlinearity, phase-locked loop, PLL, variable delay, variable slope.**

I. INTRODUCTION

TIME delay is often defined as the time difference between the threshold-crossing points of two clock edges. If delay is programmable by a digital code, a digital-to-time converter (DTC) results. It is a basic building block suitable for several applications, e.g., fractional-N phase-locked loops (PLL) [1]–[4], (sub-)sampling oscilloscopes [5], [6], automatic test equipment (ATE) [7], direct digital frequency synthesis (DDFS) [8], polar transmitter [9], radar [10], phased-array system [11], and time-interleaved ADC timing calibrations [12]. This paper aims at improving the time resolution and linearity of a DTC. A nominal full-scale delay in the order of 100 ps is targeted with fine delay steps of less than 100 fs.

The basic element of a DTC is a variable-delay element, and there are different ways to implement delay in CMOS. A distributed circuit such as an ideal transmission line can theoretically provide true time delay while keeping the waveform undistorted. However, it requires unpractically long line length in CMOS technology (e.g., $100\text{ ps} \times 2 \cdot 10^8\text{ m/s} = 20\text{ mm}$). Moreover, as CMOS interconnect losses are high and frequency dependent, different amplitudes and waveforms result at different delay tap-points along a transmission line, which introduces zero-crossing variations when sensed by a comparator [13]. Lumped circuits such as all-pass filters can approximate a true time delay compactly [14], [15] and maintain signal waveform, but noise and dynamic range are compromised.

If the waveform is not important and delayed clock generation is the purpose, digital circuits can be used. Minimum digital gate delays are on the order of 10 ps in 65 nm CMOS. However, if the difference between two gate delays is used, or if the gate delay is tunable, much smaller delay steps can be realized, for example in the order of 100 fs as will be presented in this work. Although the *absolute* delay is still limited by the intrinsic gate delay, the *relative* delay steps can be much smaller.

If delay tuning is linear, a high-linearity DTC can be realized. A linear DTC is favored, as calibration of only two points is sufficient, in contrast to a non-linear DTC that require multipoint calibration [16]. To characterize linearity, integral nonlinearity (INL) is an important metric for a DTC, similar to digital-to-analog converters (DAC). Non-zero DTC INL limits the achievable spur level in fractional-N PLLs [1]–[3], [16] and the timing accuracy in sampling oscilloscopes.

A DTC often exploits a voltage ramp generated by a current source charging a capacitor, and a comparator with threshold voltage $\mathrm{V_d}$ defining a time delay $\mathrm{t_d}$ (see Fig. 1). Switched capacitors [1], [2], [4] or switched current sources [3], [6], [7] can be applied to program delay. These approaches produce a delay by varying the slope from one ramp to another, which we refer to as the *variable-slope* method (see Fig. 2(a). Using this method, 300 fs delay resolution has been achieved in [2]. However, high resolution does not necessarily mean high linearity. In this paper, we propose a *constant-slope* method in which all ramps ideally would have the same slope, in contrast to the variable-slope method (see Fig. 2(b)). To still realize variable delay, a variable start voltage is used which can linearly program delay. We will show that this method is intrinsically more linear, allowing for a more linear DTC than variable slope offers. Before we do this in the next section, we first briefly discuss related previous work.

Manuscript received October 23, 2014; revised December 30, 2014; accepted March 07, 2015. Date of publication May 14, 2015; date of current version May 22, 2015. This paper was approved by Associate Editor Boris Murmann. This work was supported by the Dutch government through project funding in the frame of the SPITS Kennerswerkersregeling Project. Chip fabrication was sponsored by NXP Semiconductors.

J. Z. Ru was with the University of Twente, Enschede, The Netherlands, and is now with Qualcomm, Irvine, CA 92618 USA (e-mail: zhiyu.ru@gmail.com).

C. Palattella, E. A. M. Klumperink, and B. Nauta are with the University of Twente, Enschede 7500 AE, The Netherlands

P. Geraedts was with the University of Twente, Enschede, The Netherlands, and is now with Teledyne DALSA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSSC.2015.2414421

0018-9200 © 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

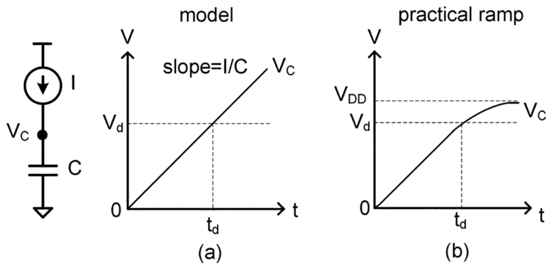


Fig. 1. A voltage ramp generated by a current source charging a capacitor, and delay $\mathrm{t_d}$ defined by the ramp crossing $\mathrm{V_d}$: (a) an ideal model; (b) a practical case.

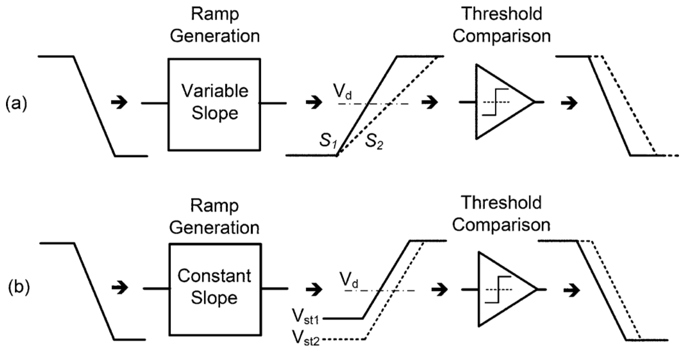


Fig. 2. Principle of (a) variable-slope method in which different slopes at comparator input define delay; (b) constant-slope method in which different start-voltages at comparator input define delay.

In [17], the nonlinearity of the variable-slope method was observed but not explained. The use of a high-gain comparator to improve INL was proposed in [17], but no measurement results were reported.

In [5], delay is controlled by tuning the threshold voltage of a comparator, which would result in linear delay control if the slope of the ramp is constant over the threshold tuning range. Practically this is challenging, as the current produced by a current source as shown in Fig. 1 depends on the voltage across it, and hence on the capacitor voltage $\mathrm{V_C}$. Moreover, the comparator in [5] works at a varying common-mode voltage, leading to a varying speed of the comparator, i.e., an extra INL source.

Another way to realize variable delay is by phase interpolation, which can be implemented using current sources [18], [19], resistors [20], [21] or delay lines [22]. The basic concept of interpolation and example waveforms are shown in Fig. 3, where the middle parts are constant-slope, assuming $\mathrm{V_A}$ and $\mathrm{V_B}$ have the same slope. However, phase interpolation is functionally different as it requires *two* edges to be present, between which it can place a new edge. In contrast, this work aims to produce a delayed edge after one incoming critical edge that triggers one charging process.

The main new contributions of this paper are threefold: 1) a concept to define a constant-slope method and to identify its fundamental advantages in terms of INL compared to a variable-slope method; 2) a new circuit topology in which the start voltage controls the delay of only one critical edge, leading to high linearity and low jitter; 3) measurement results demonstrating a fine resolution and a small INL, for which a new measurement method was devised.

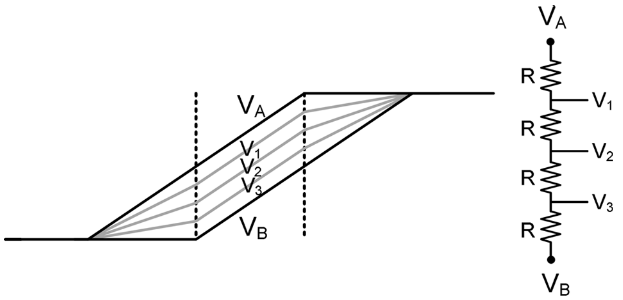


Fig. 3. Phase interpolation concept starting from two equal-slope signals $\mathrm{V_A}$ and $\mathrm{V_B}$, where the interpolated phases $(\mathrm{V_1} - \mathrm{V_3})$ have the same slope for the part within the two dashed lines.

This paper is organized as follows: Section II explains the constant-slope method and its advantage in linearity; Section III describes the design of a DTC circuit using this method; Section IV presents measurements and Section V conclusions.

II. Constant-Slope Method

A. Constant-Slope Ramp Generation

To generate a voltage ramp with a controlled slope $(S = \Delta v/\Delta t)$, often a current is used to charge a capacitor as shown in Fig. 1, where $S = I/C$. The delay time $(\mathrm{t_d})$ of this ramp from zero voltage to the voltage $\mathrm{V_d}$ is:

$$t_d = \frac{V_d}{S}. \tag{1}$$

As shown by (1), if we want one variable to control delay, we can either vary the slope S ("variable slope") at fixed voltage $\mathrm{V_d}$, or keep the slope fixed and vary voltage $\mathrm{V_d}$. In practice, however, a single ramp often has a changing slope as shown in Fig. 1(b), therefore varying $\mathrm{V_d}$ does not always give a linearly-controlled delay.

Instead, we can vary the start-voltage $\mathrm{V_{st}}$ as shown in Fig. 4 between 0 and $\mathrm{V_{st,max}}$. To generate a linearly-controlled delay, it is sufficient if the part below $\mathrm{V_{st,max}}$ is constant-slope, while for the part above $\mathrm{V_{st,max}}$ it suffices to have a constant-shape.[1] As the trajectory above $\mathrm{V_{st,max}}$ is shared for all ramps and adds a fixed amount of delay, it does not affect the linearity of the delay control function. Similarly, the same ramp start-up behavior between $\mathrm{t_0}$ and $\mathrm{t_1}$ adds a delay offset to all ramps which does not hurt linearity either.

A constant-shape above $\mathrm{V_{st,max}}$ ensures that at different $\mathrm{V_d}$, the delay τ between two ramps keeps constant, and it also renders INL benefits as described below.

B. Advantage of Constant-Slope Method on INL

We will use simple models to gain intuitive understanding. The delay function in Fig. 2 contains two distinct actions: 1) ramp generation and 2) threshold comparison. The ramp generation produces a ramp with controlled slope, while the

[1] Being constant-shape between two ramps is equivalent to having the same (local) slope at equal ramp voltage. This property allows for an alternative but important interpretation of the name "constant slope", namely that the (local) slope is constant when comparing ramps of different delay settings at equal ramp voltage.

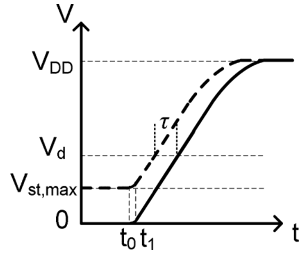


Fig. 4. Illustration of constant-slope method using practical ramps which ideally would start with a constant-slope part at least to $\mathrm{V_{st,max}}$ then share a constant-shape part beyond $\mathrm{V_{st,max}}$.

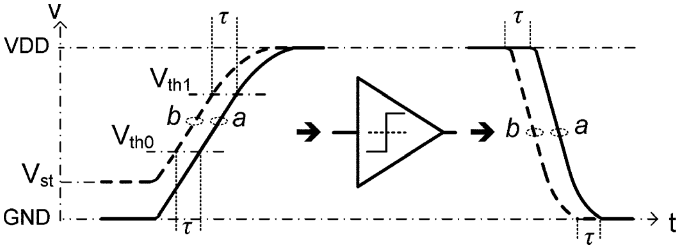


Fig. 5. Delay mechanism for constant-slope method: the start voltage $\mathrm{V_{st}}$ linearly affects delay τ, while the comparator response is identical for ramp a and b because it "sees" the same shape of both ramps.

threshold comparison defines a decision threshold V_d and produces an output edge when crossing the threshold. The variable-slope-induced INL comes from the behavior of a practical comparator.

One source of delay INL is the comparator bandwidth limit, which can be modeled by adding an RC network at the comparator output. It can be derived that, in case of an input ramp signal, the propagation delay of an RC network of any order contains nonlinear functions of the input ramp time [23], [24], e.g., exponential and logarithmic functions. Assuming linear ramp generation, the ramp time would vary linearly with code, however the propagation delay would vary nonlinearly with code due to its nonlinear function versus input ramp time in an RC network. Since poles are ubiquitous in circuits related to parasitic resistance and capacitance in transistors and interconnects, this is a source of INL in a DTC.

Another source of delay INL can be explained by the example of using an inverter as comparator. The nonlinear relationship between an inverter's delay and its input ramp time has been modeled in [25] by equations using empirical parameters obtained from simulation fitting. Three operating modes were distinguished in an inverter's response to an input ramp signal: overshoot recovery, short circuit, and output discharge [25]. During overshoot recovery, the output recovers from overshoot due to an initial input switching event; the short-circuit mode occurs when both the PMOS and NMOS conduct (but with different currents so non-zero output slope), resulting in "short circuiting" of the supply; the output-discharge mode refers to the mode with only the NMOS on. For different input slopes, the three modes contribute differently to the output transition time, which is another source of INL in a DTC. This mechanism applies to any comparator that passes through different operating modes during its input and output transitions.

To avoid the INL error associated with variable slope, we propose the constant-slope method in which the ramps keep a constant shape above $\mathrm{V_{st,max}}$, whose effect on a comparator is modeled in Fig. 5. Two rising ramps a and b at the input of the comparator have different start voltages but the same shape above $\mathrm{V_{st}}$. A delay difference τ is sensed by the comparator to produce two corresponding falling edges at the output.

Actually, the output of a practical comparator responds to a range of input voltages and modelling it as a simple comparator with one exact threshold is somewhat simplistic. Instead of a threshold, it is perhaps better to talk about a "comparator input window", for example between $\mathrm{V_{th0}}$ and $\mathrm{V_{th1}}$ in Fig. 5. When the input voltage rises to $\mathrm{V_{th0}}$, the output voltage begins to change as the comparator starts to discharge the output node (a comparator with inverted output is assumed here).

If $\mathrm{V_{st}} < \mathrm{V_{th0}}$, the start-voltage part of the ramp does not affect the comparator response. Only the part of the ramp within the input window affects the output. In Fig. 5, if the ramps a and b have the same shape between $\mathrm{V_{th0}}$ and $\mathrm{V_{th1}}$, they evoke the same response at the output. Hence their propagation delays through the comparator are equal, and both edges would also have a constant shape at the output of the comparator, no matter what the bandwidth is. Therefore the time shift between two edges at the output is the same as at the input, and there is ideally no error. Furthermore, unlike the variable-slope case, because all edges have the same shape also at the output, INL errors are also avoided in later stages, e.g., buffer stages that further steepen the output edge.

If a comparator passes through different operating modes during its transition as modeled in [25], the constant-slope method still renders benefits in INL. The reason is again that, apart from a different start voltage, both ramps in Fig. 5 have the same shape within the critical input window of the comparator whose response to both ramps is then very similar. Hence each operating mode, e.g., the short circuit or output discharge modes discussed above, renders the same contribution to the output edge for ramp a and b. The overshoot depends on the start-voltage level, however as long as the start voltages are well below the "comparator input window" then the overshoot at the output can recover before the input reaches that window, so its contribution to delay can be negligible.

To intuitively summarize, any (correlated) differences between ramps at the comparator input tend to cause INL error. The proposed constant-slope method modulates delay by changing the start voltage while keeping the critical threshold part of a ramp unaltered. Thus all ramps have the same shape in the "comparator input window", leading to the same propagation delay through a comparator or buffer stage, which minimizes INL errors. In variable-slope method, the slope is different among all ramps in the "comparator input window," so it may introduce significant INL through comparator as shown in the next sub-section.

C. Simulation

The setup in Fig. 6 is used to simulate the INL for an ideal input ramp signal. Circuit simulations were done in a 65 nm CMOS technology at 1.2V supply. In clocking applications, a

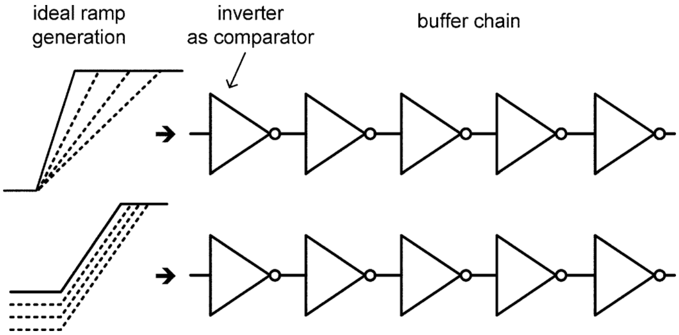


Fig. 6. INL simulation bench to compare the variable and constant slope methods.

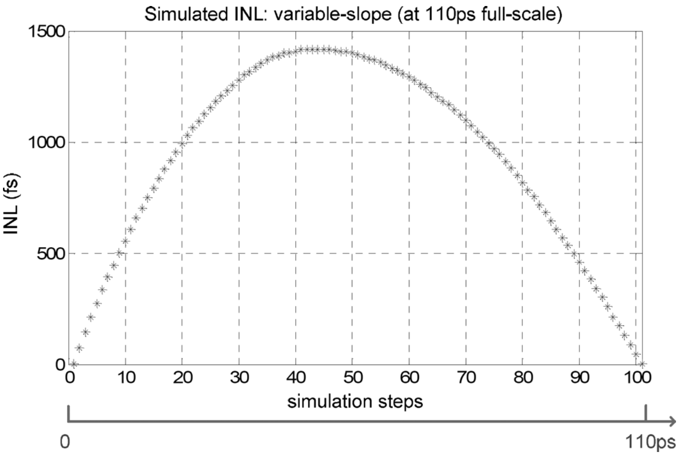


Fig. 7. Simulated INL for the variable-slope method with 110 ps full-scale delay: 1.4 ps maximum INL results.

simple inverter is often used to implement the threshold comparison and can also act as a buffer to produce a steep output edge. In Fig. 6, after the first inverter acting as a comparator, a four-stage inverter chain of identical inverters is applied to boost the slope to values close to the technology-dependent speed limit (e.g., 50 ~100 GV/s in 65 nm). This is for instance desired in sampler or phase detector applications, to precisely define the timing.

To simulate the variable-slope case, 100–300 ps rise time from GND to VDD was used, ideally resulting in 50–150 ps delay at half-VDD comparator threshold, so 100 ps delay-control range. The simulation results are shown in Fig. 7 which have an actual range of 110 ps, as the inverter threshold is not exactly half-VDD. 100 steps were taken over the whole delay range and a maximum INL of 1.4 ps is found, which is in the same order of magnitude as results found in literature [1], [2], [7].

For the constant-slope case a variable start-voltage range from 0 to 0.2 V was used (motivated later in this paper). Mapping this range to a 100 ps delay, the rise time is 600 ps from 0 V to 1.2 VDD and 500 ps from 0.2 V to 1.2 VDD. Simulation results in Fig. 8 show a maximum INL of only 15 fs, about two orders of magnitude lower than for variable slope. This clearly demonstrates the INL advantage of the constant-slope method, via the example of a simple inverter as a comparator.

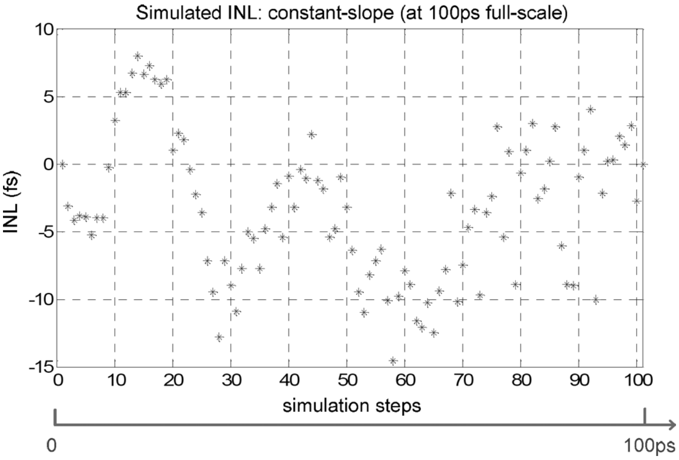


Fig. 8. Simulated INL for the constant-slope method at 100 ps full-scale delay: only 15 fs maximum INL is found compared to 1.4 ps in Fig. 7.

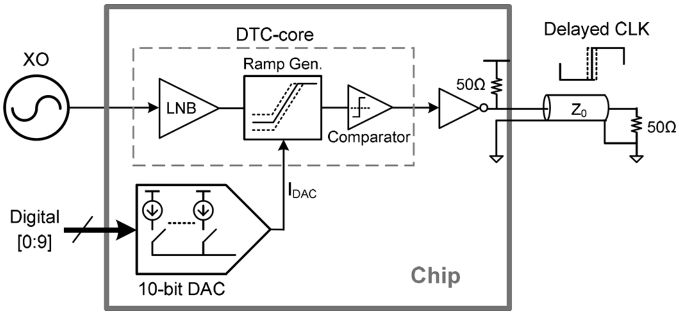


Fig. 9. Block diagram of the implemented DTC (LNB = Low-Noise Buffer).

III. DTC Demonstrator Chip Design

To demonstrate good INL in practice, a DTC chip has been implemented in 65 nm CMOS with 1.2 V supply. Fig. 9 shows the block diagram of the chip. The DTC-core is a voltage-controlled variable-delay element, which consists of a low-noise buffer (LNB), a ramp generator, and a threshold comparator. The amount of delay is controllable by a 10 bit DAC. The DTC is driven by a sine-wave from a crystal oscillator (XO), and its output delivers a rectangular-wave clock with a variable delay. An inverting buffer with 50 Ω output impedance drives the off-chip transmission line for measurements. We will discuss the design of the main blocks in the following subsections, and will also discuss INL error sources.

A. DTC-Core

Fig. 10 shows the schematic of the DTC-core. Its sub-blocks are discussed below.

1) Low-Noise Buffer: The low-noise buffer converts a sine-wave into a rectangular-wave with low added jitter. The noise of the first stage is critical given the relatively low slew-rate of a sine-wave from a 55 MHz crystal. As only one edge is critical, big NMOS transistors are used for low noise while the PMOS is small and is controlled by its driver in such a way that simultaneous conduction of the PMOS and NMOS is reduced [26]. The driver (D1) of the PMOS is shown in Fig. 11, which produces a small duty cycle therefore low supply “short-circuit” current. The big “poor man's cascode” NMOS in Fig. 11 (sized

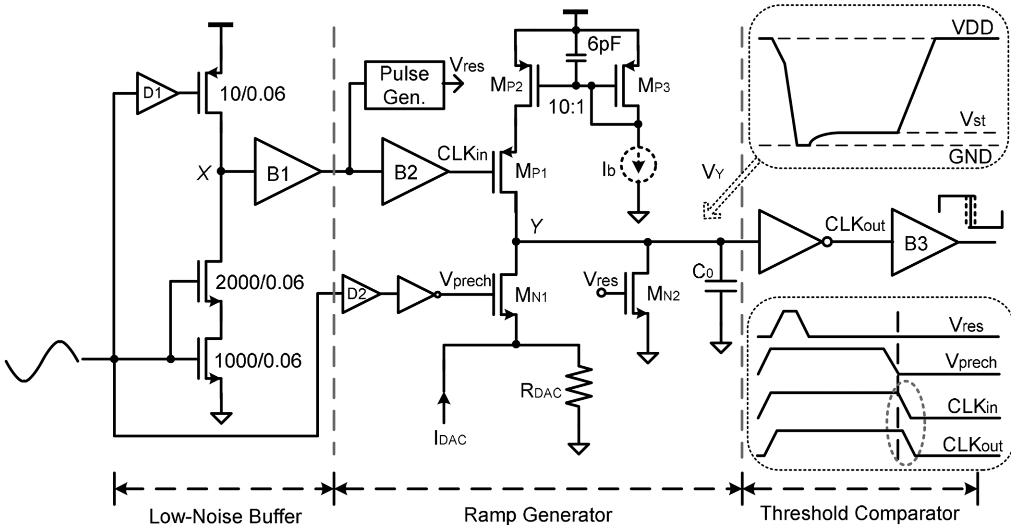


Fig. 10. Circuit schematic of the DTC-core: B1/B2/B3 are buffers made of two inverters all using regular-$\mathrm{V_{th}}$ MOSFET; D1/D2 are drivers made of two inverters using a mix of regular-$\mathrm{V_{th}}$ and high-$\mathrm{V_{th}}$ MOSFET as shown in Fig. 11.

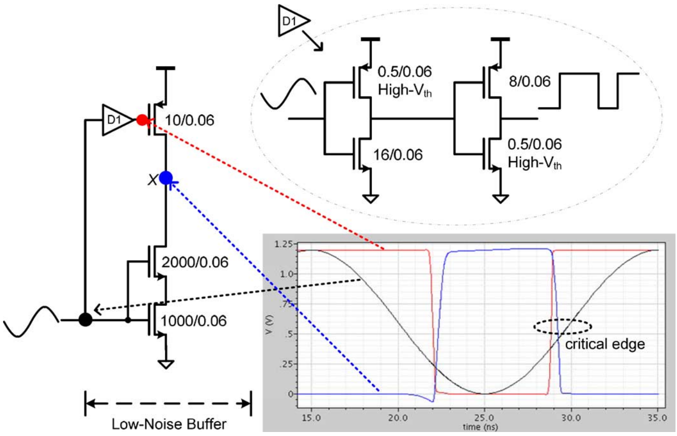


Fig. 11. PMOS driver D1 consisting of two inverters using a mix of regular-$\mathrm{V_{th}}$ and high-$\mathrm{V_{th}}$ MOSFET to produce $\sim$ 1/3 duty cycle; input and output waveforms of the low-noise buffer showing its PMOS is only on for $\sim 1/3$ duty cycle when the lower two NMOS FETs are off.

2000/0.06) helps to boost the output impedance, without requiring a dedicated bias voltage. Therefore the voltage gain around the zero-crossing points of the input rising edges is increased and so is the falling-edge steepness at the output node X, which benefits timing jitter.

2) Ramp Generator: The core of the ramp generator in Fig. 10 consists of $\mathrm{M_{P1}}$–$\mathrm{M_{P3}}$ that produce the charging current to capacitor C_0 to realize a ramp voltage. In every cycle of the DTC, node Y is first reset to GND via $\mathrm{M_{N2}}$, then pre-charged to $\mathrm{V_{st}}$ via $\mathrm{M_{N1}}$ ($V_{st} = I_{\mathrm{DAC}} \cdot R_{\mathrm{DAC}}$), after which a ramp takes place. The timing of the reset and pre-charge of $\mathrm{C_0}$ and the ramp is controlled by three signals which are all derived from the same input: $\mathrm{V_{res}}$, $\mathrm{V_{prech}}$ and $\mathrm{CLK_{in}}$. Produced by LNB with two buffers, $\mathrm{CLK_{in}}$ delivers the critical edge that activates $\mathrm{M_{P1}}$ to start the ramp. The driver D2 producing $\mathrm{V_{prech}}$ is the same as D1 shown in Fig. 11. The pulse generator producing $\mathrm{V_{res}}$ is made of an AND gate with two inputs whose delay difference defines the pulse width as shown in Fig. 12.

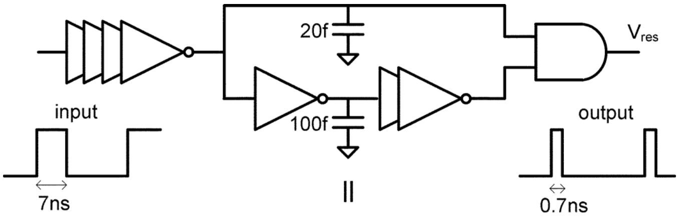


Fig. 12. Pulse generator producing $\mathrm{V_{res}}$ used in Fig. 10 with nominal 0.7 ns width; the 20 fF contributes to τ_1 which helps $\mathrm{V_{res}}$ to fit in the overall timing plan while the 100 fF contributes to τ_2 which determines the $\mathrm{V_{res}}$ pulse width.

$\mathrm{M_{P2}}$ and $\mathrm{M_{P3}}$ form a current mirror with 10:1 ratio to create a charging current, derived from an external bias current $\mathrm{I_b}$ for flexibility. The charging current can be up to a few mA, but the average current consumption is on the order of a few hundred μA because $\mathrm{M_{P2}}$ only draws current during the ramp, which is only a small fraction of the clock period. A 6 pF capacitor to VDD helps keep the gate voltage of $\mathrm{M_{P2}}$ stable and so its current. Then the different start voltages $\mathrm{V_{st}}$ at node Y have much reduced effect on $\mathrm{M_{P3}}$ and $\mathrm{I_b}$. $\mathrm{M_{P1}}$ acts as switch that starts the ramp, but also as cascode transistor to improve the output resistance of the current source $\mathrm{M_{P2}}$, and hence the linearity of the ramp voltage.

Because delay is defined as $\Delta t = \Delta v/(I/C)$, for a linear delay, it is important to realize a constant current source and capacitance, during the first part of the ramp that defines delay via different $\mathrm{V_{st}}$ values. At a 1.2V supply, this requirement limits the $\mathrm{V_{st}}$ in our circuit from GND up to about $\mathrm{V_{st,max}} = 200\,\mathrm{mV}$, where $\mathrm{M_{P1}}$ still remains well in saturation and acts well as cascode. When a ramp goes beyond $\mathrm{V_{st,max}}$, $\mathrm{M_{P1}}$ would gradually enter the triode region, however this does not cause INL, because it is a common effect, i.e., the same waveform is shared for all ramps beyond $\mathrm{V_{st,max}}$.

The linear poly resistor $\mathrm{R_{DAC}}$ defines $\mathrm{V_{st}}$ based on current $\mathrm{I_{DAC}}$. The nonlinear switch resistance $\mathrm{R_{on}}$ of $\mathrm{M_{N1}}$ has negligible contribution to INL if $\mathrm{R_{DAC}} \gg \mathrm{R_{on}}$, which can be understood by applying Thévenin's theorem where $\mathrm{R_{DAC}}$ and $\mathrm{R_{on}}$ are in series. The linear metal capacitor $\mathrm{C_0}$ in parallel to the nonlinear parasitic capacitance at node Y defines the slope of the voltage ramp. This improves linearity as the combined capacitance is less dependent on voltage.

Given a supply voltage, the usable start-voltage range is limited. If the voltage range is fixed, to achieve a larger delay requires a lower slope (i.e., slower ramp), posting a delay-jitter trade-off.

3) Threshold Comparator: The threshold comparator uses a simple inverter to sense the ramp voltage created at node Y. The nominal threshold voltage of the inverter was designed to be around half-VDD (~ 600 mV), which is much larger than the 200 mV maximum $\mathrm{V_{st}}$. Buffer B3 (two scaled-up inverters) steepens the output edges.

B. Digital-to-Analog Converter

To save design time, an existing 10 bit current-steering DAC IP-block is co-integrated on the same chip. The segmented DAC is divided into two sub-DACs, a 5 bit binary-weighted sub-DAC for LSBs and a 5 bit unary-weighted sub-DAC for MSBs. For its performance, we rely on the specification datasheet, which however is not very detailed. Hence, we resort to calculations and estimations to derive some of the specifications.

The DAC is specified to operate at 2.5 V supply, but it can also operate at 1.2 V with a more limited output voltage range. The DAC specification indicates a maximum INL of ±2 LSB (±0.2%) at an output range of 0–800 mV and a 400 MHz speed. The INL should improve when only 0–200 mV output range and 55 MHz speed are used in this design.

The DAC noise is not found in the IP's datasheet. A first-order calculation was done assuming the thermal noise is dominant at a low switching speed of 55 MHz. For a current-steering DAC, its current noise can be modelled as:

$$\begin{aligned}{\sigma_{i_n}}^2 &= 4kT \cdot g_{m,DAC} \cdot \Delta f \\ &= 4kT \cdot \frac{2I_{\mathrm{DAC}}}{V_{\mathrm{eff}}} \cdot \frac{1}{4R_{\mathrm{DAC}} \cdot C_0} \\ &= \frac{2kT \cdot I_{\mathrm{DAC}}}{V_{\mathrm{eff}} \cdot R_{\mathrm{DAC}} \cdot C_0} \end{aligned} \tag{2}$$

where Δf is the equivalent DAC noise bandwidth defined as $1/(4R_{\mathrm{DAC}}C_0)$ with R_{DAC} and C_0 in Fig. 10, and V_{eff} is the overdrive voltage of the DAC current sources.

The DAC current noise is converted to voltage by R_{DAC} and then produces timing jitter, which can be modelled as:

$$\sigma_t = \frac{\sigma_{i_n} \cdot R_{\mathrm{DAC}}}{\Delta v/\Delta t} = \frac{1}{\Delta v/\Delta t} \cdot \sqrt{\frac{2kT \cdot I_{\mathrm{DAC}} \cdot R_{\mathrm{DAC}}}{V_{\mathrm{eff}} \cdot C_0}}. \tag{3}$$

In our design, $\Delta v/\Delta t$ is equal to 200 mV/100 ps; $I_{\mathrm{DAC}} = 0.87$ mA; $R_{\mathrm{DAC}} = 200\ \Omega$; $C_0 = 1.3$ pF; $kT = 4.1\text{e-}21$J. Even assuming a rather low $V_{\mathrm{eff}} = 150$ mV, the result of (3) is 43 fs, i.e., < 0.5 LSB for a 10 bit DAC and a 100 ps full-scale delay. Note that this is at the maximum DAC output current, i.e., the worst-case noise.

C. 50 Ω Output Buffer

To be able to measure the DTC, an output buffer is designed which includes an inverter and an integrated 50 Ω pull-up resistor (see Fig. 9). When connected to an off-chip cable and equipment with 50 Ω to ground, the buffer output establishes a DC bias voltage nicely around half-VDD. The inverter is sized to provide around 0.6 $\mathrm{V_{pp}}$ swing at the matched output so that the variation of inverter output resistance does not have much effect on the output impedance matching.

D. Error Sources

A DTC is usually meant to produce a well-controlled amount of delay, however non-idealities such as noise, distortion, process-voltage-temperature (PVT) variations, and mismatches introduce timing errors.

For a switching circuit such as a DTC, the jitter is often lower with higher signal slope or larger transistor size and current. The jitter of the implemented DTC is dominated by the LNB due to the low sine-wave slope at 55 MHz input.

Other than jitter, the timing error of a DTC can be divided into offset, linear and nonlinear errors. An offset error means a common delay shift to all delay steps while the relative delay from one step to another remains unchanged; a linear error means that all delay steps are scaled by the same ratio, i.e., full-scale delay changes but delay steps are still equal and there is no DNL or INL; nonlinear errors render a code dependent step size, leading to DNL/INL.

If the threshold voltage of an inverter varies over PVT, in a variable-slope method, this will cause offset and linear errors (so the full-scale) which can be seen in Fig. 2 by moving $\mathrm{V_d}$ up and down, but potentially also different INL. For instance, INL in percentage changes if nonlinear errors scale differently than the full-scale delay does. On the other hand, a $\mathrm{V_{th}}$-shift means that the inverter characteristic changes, which often also leads to INL change. Furthermore, a practical ramp is not a perfectly straight line and its slope is different at different voltage levels (see Fig. 1(b)). When a ramp passes through an inverter for which PVT changes the threshold, the inverter sees a different slope, so the nonlinear effects change, and INL values change. Instead, for the constant-slope method only an offset in delay occurs, which can be seen in Figs. 2 and 5. It does not cause linear errors because the delay from one ramp to another is the same at any voltages. As explained in Section II, it also intrinsically does not cause nonlinear errors by comparator no matter what the threshold is.

As the constant-slope method minimizes the INL associated with comparator, the remaining error sources are mostly in ramp generation, which can be minimized by design.

In practice, the start-up behavior of a ramp is not instantaneous, but rather has an initial over-shoot due to capacitive coupling and a rounded start-up waveform because a charging current is not fully turned on instantaneously. Simulation indicates that this effect is largely independent of the start-voltage for the used 0–200 mV $\mathrm{V_{st}}$-range. This mainly adds an offset to the delays.

A rising slope at node Y in Fig. 10 can be written as

$$S = \frac{10 \cdot I_b}{C_0} = \frac{V_{st}}{\Delta t} = \frac{I_{\rm DAC} \cdot R_{\rm DAC}}{\Delta t}. \quad (4)$$

Equation (4) can be re-arranged to derive the delay Δt:

$$\Delta t = \frac{V_{st}}{10 \cdot I_b} \cdot C_0 = \frac{I_{\rm DAC}}{10 \cdot I_b} \cdot R_{\rm DAC} \cdot C_0. \quad (5)$$

Equation (5) shows this delay is proportional to an RC time constant which is subject to PVT variations, introducing a linear error. The DAC current and the bias current can be derived from the same current source, so PVT variations are removed by taking the current ratio. However, any mismatch in current mirrors will introduce linear errors. Many applications will require the delay range to be aligned to another clock, e.g., a VCO period in a PLL, which will then also calibrate these linear errors.

The INL for the constant-slope design is related to the circuit nonlinearity caused by the varying $\rm V_{st}$, including the nonlinearity of current source $\rm M_{P2}$, the nonlinearity of junction capacitance at node Y, and the nonlinearity of pre-charge switch $\rm M_{N1}$. The former two affect the I/C ratio and therefore the slope, while the latter two affect the settling of $\rm V_{st}$ due to nonlinearity in the RC time constant. Note that these nonlinearities have been largely reduced by measures discussed in Section III-A2), including cascode $\rm M_{P1}$, linear $\rm C_0$, and linear $\rm R_{DAC}$. Furthermore, a relatively small $\rm V_{st}$ range of 200 mV helps limit these nonlinear effects; also, a 55 MHz operating frequency gives enough settling time to reduce the settling error of $\rm V_{st}$.

Another source of INL comes from the DAC. The mismatch of the DAC cells and the nonlinearity of its output impedance affect the DAC INL and therefore $\rm V_{st}$, directly translating to the delay INL as shown in (5). Note that the DAC INL similarly hurts the delay INL in a variable-slope method. This effect relates to the DAC design, and is not intrinsic nor distinctive for the constant or variable slope method.

E. Simulation Results

Using a PSP Model, the circuit in Fig. 10 has been simulated together with an ideal DAC producing 0–200 mV as $\rm V_{st}$. The input of the LNB is driven by a 55 MHz sine-wave with 1.2 $\rm V_{pp}$ swing.

The DTC INL is defined similarly to that of a DAC: assuming the total number of bits is N, the INL at digital code k is then defined as

$$INL(k) = \tau(k) - \frac{k}{2^N - 1} \cdot \tau_{\rm FS} \quad (6)$$

where $\tau(k)$ is the measured delay at code k and $\tau_{\rm FS}$ is the measured full-scale delay.

The simulated INL is shown in Fig. 13 with 200 simulation steps at 100 ps full-scale. The maximum INL error is less than 50 fs (0.05%) and mainly due to non-ideality in the ramp generation such as the residual current-source nonlinearity. Some uncertainty in results is likely due to simulation accuracy at such small time resolution (note that the pattern is rather regular).

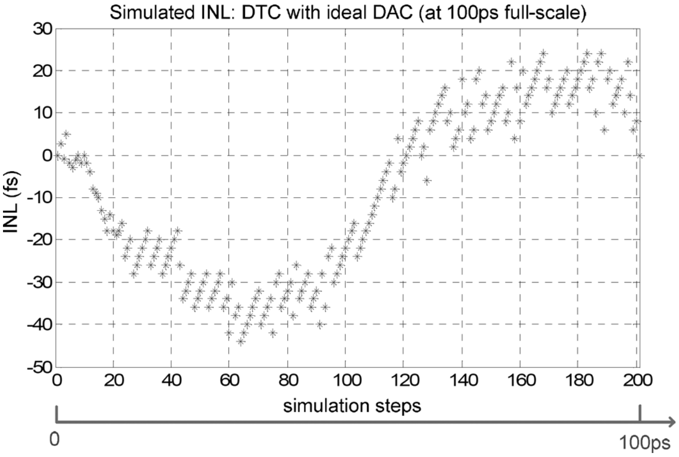


Fig. 13. Simulated INL of the DTC-core at 100 ps full scale, with ideal DAC, using a PSP model for the 65 nm CMOS transistors.

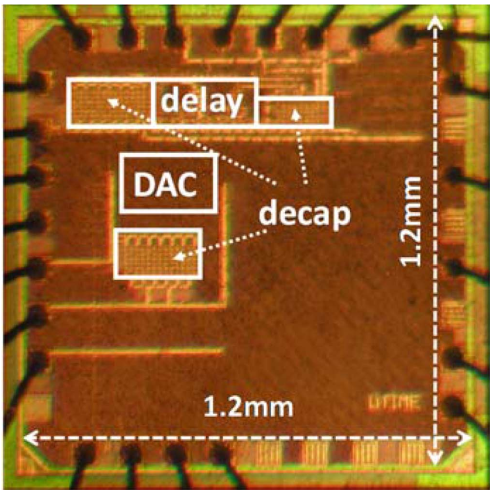


Fig. 14. Chip photo of the DTC realized in 65 nm CMOS with active area of 0.1 mm^2.

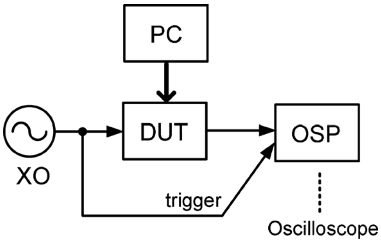


Fig. 15. Setup for INL measurements in the time domain using an oscilloscope.

The RMS jitter was also simulated within a bandwidth up to half of the clock rate, resulting in 109 fs and 99 fs at $\rm V_{st}$ of 0 mV and 200 mV, respectively. Lower jitter at higher $\rm V_{st}$ is due to the smaller charging time so less noise integration. The LNB alone is simulated to have a jitter of 81 fs, which is the biggest contribution due to its low-slope sine-wave input.

IV. MEASUREMENTS

The chip design shown in Fig. 9 was fabricated in 65 nm CMOS and a chip photo is shown in Fig. 14. The active area of the DTC-core and DAC is about 0.1 mm^2, each taking roughly

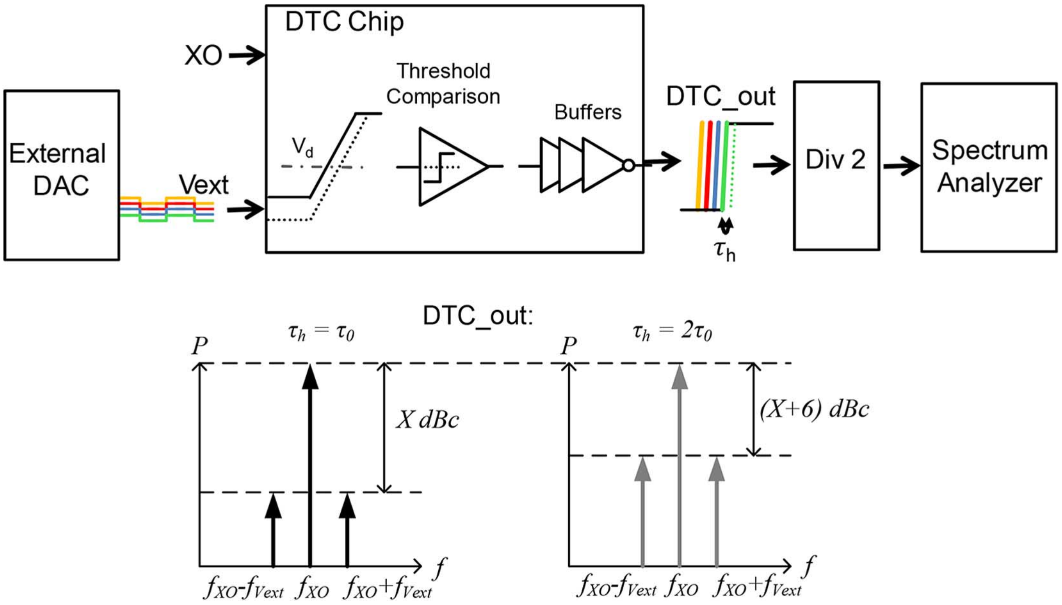


Fig. 16. Setup for INL measurements in the frequency domain using the method of [27].

half. The chip is packaged in a 32-pin Heat-sink Very-thin Quad Flat-pack No-leads (HVQFN) package. All measurements were performed on PCB.

A. Delay INL

Fig. 15 shows a simplified setup used to measure DTC delay and INL in our time-domain experiment. The chip (DUT) receives a 1.2 Vpp sine-wave input from the 55 MHz crystal oscillator (XO) and delivers a 0.6 Vpp rectangular-wave output to a 50 Ω $\mathrm{Z_{in}}$ oscilloscope (OSP). The crystal signal is also used as reference to trigger the sampling oscilloscope. The chip is programmed from a computer via an integrated two-pin serial-bus interface. Using this setup, we estimated the deterministic part of the INL to be in the order of 150 fs at 102 ps full-scale delay (0.15%) and 250 fs at 304 ps full-scale (0.08%). However, the results contain large measurement uncertainties on the same order as the estimated INL therefore it is difficult to assess the reliability of these measurements and draw conclusions.

Because the time-domain method is not good enough to directly measure the INL of the chip, we developed an indirect method for the characterization of the DTC-core, that avoids the oscilloscope and instead uses a spectrum analyzer. The basic idea is to periodically modulate the delay of the DTC between two distinct values, which results in a spur [27]. Such a spur can be measured with high fidelity in the frequency domain, as only noise and interference in a small frequency band around the spur frequency will pollute the results. In contrast, a sub-sampling oscilloscope is wideband, and hence sensitive to noise and interference in a wide band.

The proposed measurement setup is shown in Fig. 16. The on-chip DAC is off as its serial digital interface is too slow for the modulation frequency. Instead, an external DAC (Agilent M8190 A Arbitrary Waveform Generator) was used to produce a square-wave voltage $(\mathrm{V_{ext}})$ that switches slowly compared to the input XO $(f_{\mathrm{Vext}} = 2.5\ \mathrm{MHz}, f_{\mathrm{XO}} = 55\ \mathrm{MHz})$. We used 10 bits as our full-scale out of the DAC's 14 bit maximum range. We measured the external DAC performance and found that its INL is below ±0.5 LSB (0.05% referring to 10 bit full-scale) which is not the bottleneck in our DTC-INL measurement.

Note that the start voltage now is defined by $\mathrm{V_{ext}}$, so each voltage level determines a position of the DTC output rising edge; a square wave at $\mathrm{V_{ext}}$ produces a delay/phase modulation at the DTC output, because its rising edge jumps periodically between two positions. This phase modulation appears in the frequency domain as a couple of sidebands, where the strongest occurs at an offset frequency $f_{\mathrm{V_{ext}}}$ from f_{XO} which is the carrier frequency of the DTC output (see Fig. 16). These sidebands can be measured using a spectrum analyzer. Only the rising edges of DTC output are programmable, therefore a frequency divider by 2 is inserted between the DTC chip and the spectrum analyzer, in order to discard the falling edges of the DTC output.

Just like the modulating signal $\mathrm{V_{ext}}$, the phase change of the signal at the divider output is also a square wave. By using the standard modulation theory [28], it can be shown that the relative strength of the first sideband (either on the left or right side of the carrier, see Fig. 16) in dBc is related to the delay step produced by the square wave as the following equation [27]:

$$\mathrm{spur\_dBc} = 20\log_{10}\left(\frac{\tau_h}{T_{ck}}\right) \tag{7}$$

where τ_h is the delay step of the rising edge, produced by the voltage step of the hth square wave $\mathrm{V_{ext}}$, and $\mathrm{T_{ck}}$ is the period of the DTC output.

To achieve high accuracy in spur measurements, it is beneficial to nominally always measure the same spur strength: range switching in a spectrum analyzer is avoided in that way and the nonlinearities in the power detector are minimized. In terms of DAC codes, in one code-sweep the full code range is covered with 40 identical code steps. Each code step produces a square

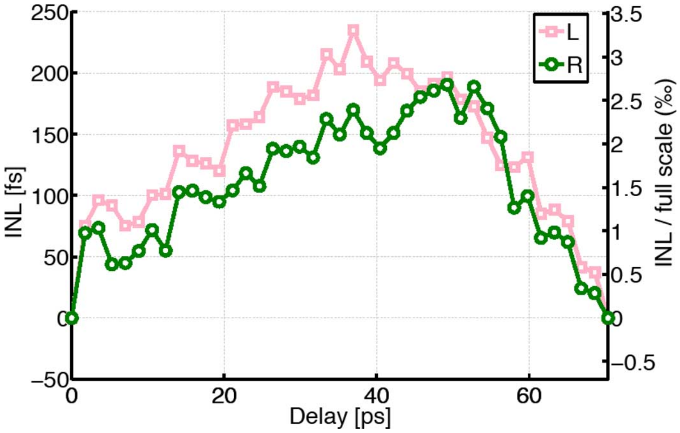


Fig. 17. Measured INL at 71 ps full-scale using the proposed method ($V_{\rm ext} =$ 0 to 64.8 mV); the L and R curves refer to measurements from the left and right spur-sidebands respectively.

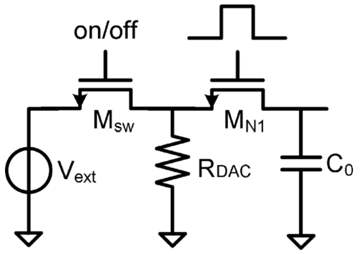


Fig. 18. Using an external voltage-mode DAC where the voltage signal $V_{\rm ext}$ goes through an extra static switch $M_{\rm sw}$ which contributes additional INL, and the linear $R_{\rm DAC}$ is not effective to reduce the nonlinearity of $M_{\rm N1}$ compared to using current-DAC.

wave in terms of $V_{\rm st}$, where the amplitude is fixed (so nominally equal delay steps and spur strength), but DC levels are increasing from one code step to the next. For each code step, we measure a spur level which is then converted to a delay step τ_h via (7). Due to nonlinearity there will be variations in the measured $\tau_h (1 \leq h \leq 40)$. These variations correspond to DNL errors which can be calculated with the following equation:

$$DNL(h) = \tau_h - \tau_{id} \tag{8}$$

where τ_{id} is the ideal delay step produced by each square wave, which is estimated as the average of all measured τ_h values for a complete code-sweep. The INL is the cumulative sum of the DNL.

Both low-frequency and high-frequency noise exist in the measurements. We chose to do each sweep (40 points) within 10 minutes, and then repeat the procedure 50 times, so 50 nominally equal data sets result. In this way, a single INL plot of each sweep is less sensitive to low-frequency noise, and an average of 50 helps to remove high-frequency noise.

The INL curve from this method for a full-scale delay of 71 ps, using 40 delay-steps, is shown in Fig. 17. Both the left and right spur-sidebands were measured and they agree within about 50 fs with each other. The two y-axes in Fig. 17 indicate, respectively, the absolute INL in femtoseconds, and its normalized value to the full-scale delay, i.e., INL in percentage. The absolute INL is within 235 fs. The normalized INL is within 0.33%, corresponding to an effective resolution of $\log_2(1 \div 0.33\%) =$ 8.2 bits, when only considering the INL-limitation.

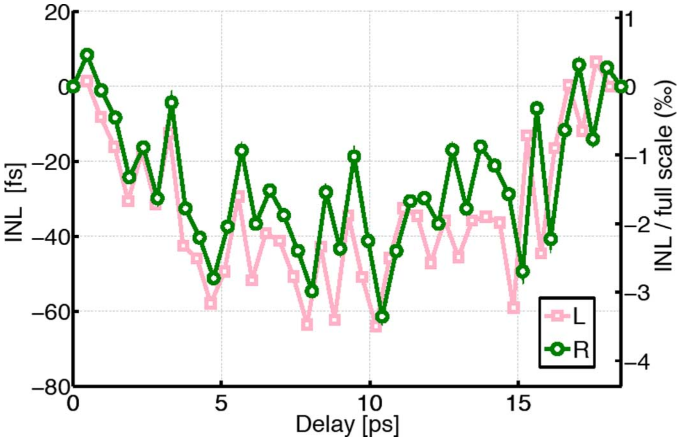


Fig. 19. Measured INL at 19 ps full-scale using the proposed method ($V_{\rm ext} =$ 0 to 33.6 mV).

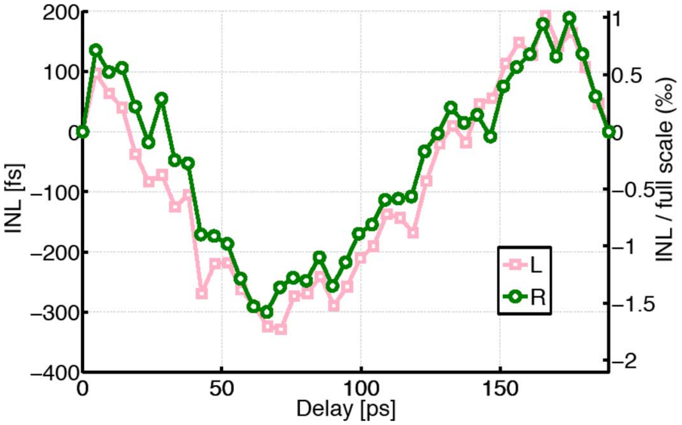


Fig. 20. Measured INL at 189 ps full-scale using the proposed method ($V_{\rm ext} =$ 0 to 101.5 mV).

The measured INL is the combination of the chip and the external DAC, while the on-chip DAC is not involved. The external DAC is a voltage-mode DAC instead of a current-DAC, therefore the linear $R_{\rm DAC}$ in Fig. 10 is not effective in this case to help the nonlinear $R_{\rm on}$ of $M_{\rm N1}$. As shown in Fig. 18, $V_{\rm ext}$ goes through an extra on-chip static switch $M_{\rm sw}$, which contributes additional nonlinearity. Experiments show that using external voltage-DAC degrades INL and limits the linear $V_{\rm ext}$ range to about 100 mV, reducing the linear delay range, compared to the case with an on-chip current-DAC.

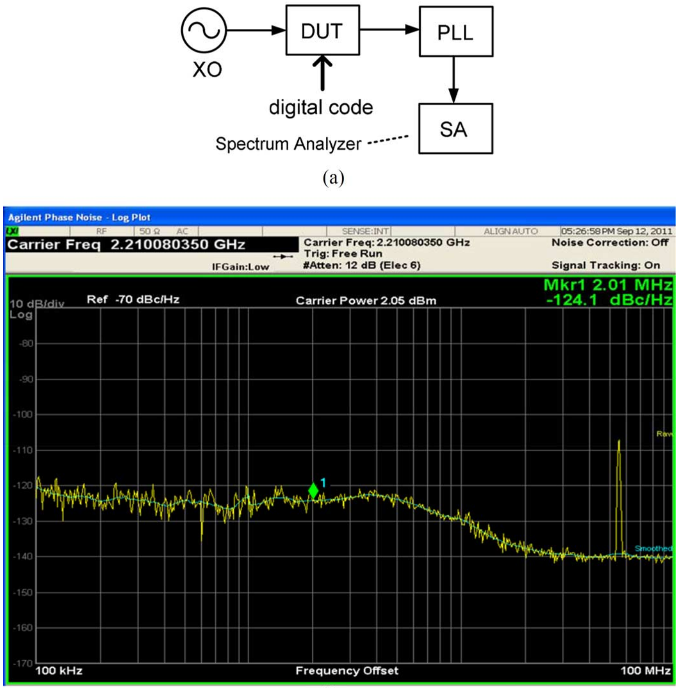
The full-scale delay is varied roughly from 20 to 200 ps. The measured INL of 19 ps and 189 ps full-scales are shown in Figs. 19 and 20, respectively. The maximum INL is 64 fs for 19 ps full-scale (normalized INL = 0.34%, 8.2 bits), and 328 fs for 189 ps full-scale (normalized INL = 0.17%, 9.2 bits), showing good linearity over a large delay range.

Very different settings are used for these different full-scales, in terms of DAC voltage range, charging current value, and charging capacitor value. The different contributions to nonlinearity (such as current source, switch resistance, parasitic capacitance, and DAC) will increase or decrease at different settings.

TABLE I
COMPARISON WITH OTHER RECENT WORK ON DTC PERFORMANCE

	This Work	[1] ISSCC11	[2] ISSCC11	[4] ESSCIRC14	[5] VLSI06	[7] ISSCC06
Delay Method	Variable Start-Voltage (Constant Slope)	Variable Slope	Variable Slope	Variable Slope	Variable Threshold	Variable Slope
Fine-Delay Range	19~189ps	186ps	247~338ps	563ps	64ps	59ps
Resolution	19~185fs	4700fs	241~330fs	550fs	1000fs	1830fs
INL	64fs@19ps (0.34%) 235fs@71ps (0.33%) 328fs@189ps (0.17%)	1900fs [3] @186ps (1%)	3000fs [3] @305ps (1%)	990fs@563ps (0.18%)	3200fs @ 64ps (5%)	3000fs @ 59ps (5%)
In-Band PN (dBc/Hz)	<-124 @ 2.21GHz [1] (-131 @ 1GHz)	<-100 @ 5.38GHz (-115 @ 1GHz)	<-102 @ 3.28GHz (-112 @ 1GHz)	-155 @ 40MHz (-127 @ 1GHz)	N/A	N/A
Jitter	<210fs [1]	<300fs	<400fs	< 250fs	N/A	700fs
Power (mW)	0.8+1.0 [1] @ 55MHz	> 0.22 [2] @ 48MHz	2.2 [2] @ 40MHz	0.5 @ 40MHz	N/A	N/A
CMOS Tech.	65nm	65nm	65nm	28nm	90nm	0.18um

[1] Noise and power data are measured at setting for ~100ps delay; [2] Power of only DTC core, no power-hungary low-noise buffer;
[3] Estimation based on fractional-spur level using analysis in [22] to show the order of magnitude of INL;



(a)

(b)

Fig. 21. (a) Setup for phase noise measurement. (b) Measured phase noise of the DTC as a reference buffer for a low-jitter PLL (reference spur at 55 MHz).

Therefore, different subtle nonlinearity mechanism can be dominant at different full-scales, so the INL shape or even polarity can change. It is difficult to exactly pinpoint all mechanisms and match them to a model. On the other hand, we repeated many of the measurements and find reproducible results, while the measured curves from the two spur-sidebands also match each other in all three cases. These results indicate that very competitive performance can be achieved.

B. Phase Noise and Jitter

As mentioned in Section III-E, the simulated jitter is about 100 fs, which is less than the jitter of the oscilloscope we used. Hence a time-domain measurement was meaningless. Since the DTC is running at the crystal frequency, direct phase-noise measurement is also challenging, as it represents a very low phase-noise level at 55 MHz carrier. Also the measurement should only be sensitive to the rising edge of the DTC output.

In an attempt to still quantify the phase noise, we used a previously published low-jitter PLL [29] as a frequency multiplier with the setup in Fig. 21(a). Within the PLL loop bandwidth, the DTC noise is conveyed to its VCO output. As the VCO runs at a much higher frequency than the reference clock (2.2 GHz versus 55 MHz), a given timing jitter corresponds to more phase variation making phase-noise measurements easier. The on-chip DAC is used in the noise measurement.

At the DTC setting with about 100 ps full-scale delay, the total measured phase noise from the DTC and the PLL together is shown in Fig. 21(b). The in-band phase noise floor at 2 MHz is -124 dBc/Hz at a 2.2 GHz carrier with less than ± 1 dB variation for all digital codes, while the PLL alone without DTC showed -125 dBc/Hz at 2 MHz [29]. This shows the DTC is suitable for low-phase-noise applications. The integrated jitter from 100 kHz to 100 MHz is 210 fs for the DTC and the PLL together, at a loop bandwidth of 5 MHz. Note that the DTC should only contribute significantly to the noise within the loop bandwidth due to the PLL low-pass transfer function from the reference path to the VCO output.

C. Benchmark

Table I compares this DTC with other recent work. This work demonstrates the finest time resolution and achieves the best INL when benchmarked at a similar full-scale delay. To evaluate a DTC design, it is more appropriate to compare INL for similar full-scale delays, because not only absolute INL but also normalized INL often changes with full-scale delay for the same DTC. At similar full-scale and in terms of normalized INL in percentage, at 71 ps delay compared to [5] and [7] the INL is 15x better; at 189 ps delay compared to [1] and [2] the INL is 6x better. A recent DTC [4] shows a similar INL in percentage (0.18%), but our work achieves this INL at a 3x smaller full-scale delay and 3x finer resolution. Note that achieving the same INL in percentage at a smaller full-scale delay is more difficult, as small absolute delay errors become more relevant. Based on information provided in Section IV-A, we expect that using the on-chip DAC would give even better INL.

This work is also competitive in terms of jitter, and certainly for in-band phase noise when applied in a PLL. At 1.2 VDD, 55 MHz input, and 102 ps delay, the power consumption of the DTC-core is 0.8 mW ($\mathrm{I_b} = 260\ \mu\mathrm{A}$) which can be lowered with process scaling, and the DAC consumes 1 mW which can be lowered by a customized design for 55 MHz speed.

V. Conclusions

This paper has shown that the popular variable-slope delay method suffers from INL due to the variable slope of the input ramp voltage in combination with bandwidth limitations and the transition through different operating modes of the threshold comparator. A constant-slope method is proposed that generates delay by varying the start voltage of a ramp instead of its slope, which strongly improves INL.

A DTC chip based on this method is implemented in a 65 nm CMOS. It receives a sine wave as input and delivers a digitally-controlled time-delayed clock edge at the output. A 10 bit DAC defines the start voltage of the critical constant-slope ramp.

The DTC INL was measured using a newly developed frequency-domain method, detecting a spur generated by modulating the DTC phase. Measurement results show that the INL is within 328 fs for 189 ps full-scale delay (0.17%) and within 64 fs for 19 ps full-scale delay (0.34%).

Acknowledgment

The authors sincerely thank G. van der Weide, N. Pavlovic, H. Brekelmans, X. He, J. van Sinderen, D. Schinkel, and R. Roovers for discussions and tape-out assistance. The authors also sincerely thank G. Wienk, H. de Vries, J. Velner, T. Aarnink, X. Gao, and M. Soer for help during the multi-year period of measurements and many discussions that finally lead to the newly developed spur-detection based measurement method.

References

[1] N. Pavlovic and J. Bergervoet, "A 5.3 GHz digital-to-time-converter-based fractional-N all-digital PLL," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 20–24, 2011, pp. 54–56.

[2] D. Tasca, M. Zanuso, G. Marzin, S. Levantino, C. Samori, and A. L. Lacaita, "A 2.9-to-4.0 GHz fractional-N digital PLL with bang-bang phase detector and 560 fsrms integrated jitter at 4.5 mW power," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2011, pp. 88–90.

[3] R. B. Staszewski *et al.*, "Spur-free all-digital PLL in 65 nm for mobile phones," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2011, pp. 52–54.

[4] N. Markulic, K. Raczkowski, P. Wambacq, and J. Craninckx, "A 10-bit, 550-fs step digital-to-time converter in 28 nm CMOS," in *Proc. Eur. Solid-State Circuits Conf. (ESSCIRC)*, 2014.

[5] K. Inagaki, D. Antono, M. Takamiya, S. Kumashiro, and T. Sakurai, "A 1-ps resolution on-chip sampling oscilloscope with 64:1 tunable sampling range based on ramp waveform division scheme," in *IEEE VLSI Symp. Dig.*, 2006.

[6] M. Safi-Harb and G. W. Roberts, "70-GHz effective sampling time-base on-chip oscilloscope in CMOS," *IEEE J. Solid-State Circuits*, vol. 42, no. 8, pp. 1743–1757, Aug. 2007.

[7] T. Okayasu, M. Suda, K. Yamamoto, S. Kantake, S. Sudou, and D. Watanabe, "1.83 ps-Resolution CMOS dynamic arbitrary timing generator for > 4 GHz ATE applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2006.

[8] S. Talwalkar, T. Gradishar, B. Stengel, G. Cafaro, and G. Nagaraj, "Controlled dither in 90 nm digital to time conversion based direct digital synthesizer for spur mitigation," in *IEEE RFIC Symp. Dig.*, 2010.

[9] Y.-C. Choi, S.-S. Yoo, and H.-J. Yoo, "A fully digital polar transmitter using a digital-to-time converter for high data rate system," in *IEEE Int. Symp. RFIT Dig.*, 2009, pp. 56–59.

[10] D. Zito, D. Pepe, M. Mincica, and F. Zito, "A 90 nm CMOS SoC UWB pulse radar for respiratory rate monitoring," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2011, pp. 40–41.

[11] Z. Safarian, C. Ta-Shun, and H. Hashemi, "A 0.13 μm CMOS 4-channel UWB timed array transmitter chipset with sub-200 ps switches and all-digital timing circuitry," in *IEEE RFIC Symp. Dig.*, 2008, pp. 601–604.

[12] K. Poulton, R. Neff, A. Muto, W. Liu, A. Burstein, and M. Heshami, "A 4 GSample/s 8 b ADC in 0.35 μm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2002.

[13] E. A. M. Klumperink, K. T. Carlo, B. Ruggeberg, and E. J. M. van Tuijl, "AM suppression with low AM-PM conversion with the aid of a variable-gain amplifier," *IEEE J. Solid-State Circuits*, vol. 31, no. 5, pp. 625–633, May 1996.

[14] J. Buckwalter and A. Hajimiri, "An active analog delay and the delay reference loop," in *IEEE RFIC Symp. Dig.*, 2004.

[15] S. K. Garakoui, E. A. M. Klumperink, B. Nauta, and F. E. van Vliet, "A 1-to-2.5 GHz phased-array IC based on gm-RC all-pass time-delay cells," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2012.

[16] S. Levantino, G. Marzin, and C. Samori, "An adaptive pre-distortion technique to mitigate the DTC nonlinearity in digital PLLs," *IEEE J. Solid-State Circuits*, vol. 49, no. 8, pp. 1762–1772, Aug. 2014.

[17] S. Alahdab, A. Mantyniemi, and J. Kostamovaara, "A 12-Bit digital-to-time converter (DTC) with sub-ps-level resolution using current DAC and differential switch for time-to-digital converter (TDC)," in *Proc. Int. Instrumentation and Measurement Technol. Conf. (I2MTC)*, 2012.

[18] T. H. Lee, K. S. Donnelly, J. T. C. Ho, J. Zerbe, M. G. Johnson, and T. Ishikawa, "A 2.5 V CMOS delay-locked loop for 18 Mbit, 500 megabyte/s DRAM," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1491–1496, Dec. 1994.

[19] S. Sidiropoulos and M. Horowitz, "A semidigital dual delay-locked loop," *IEEE J. Solid-State Circuits*, vol. 32, no. 11, pp. 1683–1692, Nov. 1997.

[20] J. van Valburg and R. J. van de Plassche, "An 8-b 650-MHz folding ADC," *IEEE J. Solid-State Circuits*, vol. 27, no. 12, pp. 1662–1666, Dec. 1992.

[21] S. Henzler, S. Koeppe, D. Lorenz, W. Kamp, R. Kuenemund, and D. Schmitt-Landsiedel, "A local passive time interpolation concept for variation-tolerant high-resolution time-to-digital conversion," *IEEE J. Solid-State Circuits*, vol. 43, no. 7, pp. 1666–1676, Jul. 2008.

[22] M. Zanuso, S. Levantino, C. Samori, and A. L. Lacaita, "A wideband 3.6 GHz digital $\Delta\Sigma$ fractional-N PLL with phase interpolation divider and digital spur cancellation," *IEEE J. Solid-State Circuits*, vol. 46, no. 3, pp. 627–638, Mar. 2011.

[23] R. van de Plassche, *CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters*, 2nd ed. Boston, MA, USA: Kluwer Academic, 2003.

[24] M. Rosario, P. Gaetano, and P. Massimo, "Propagation delay of an RC-chain with a ramp input," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 54, no. 1, pp. 66–70, Jan. 2007.

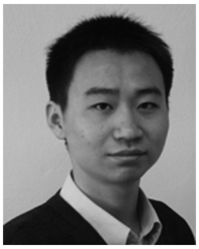
[25] J. M. Daga and D. Auvergne, "A comprehensive delay macro modeling for submicrometer CMOS logics," *IEEE J. Solid-State Circuits*, vol. 34, no. 1, pp. 42–55, Jan. 1999.

[26] X. Gao, E. Klumperink, G. Socci, M. Bohsali, and B. Nauta, "A 2.2 GHz sub-sampling PLL with 0.16 psrms jitter and -125 dBc/Hz in-band phase noise at 700 μW loop-components power," in *IEEE Symp. VLSI Circuits Dig.*, 2010.

[27] C. Palattella, E. Klumperink, Z. Ru, and B. Nauta, "A sensitive method to measure the integral nonlinearity of a digital-to-time converter, based on phase modulation," *IEEE Trans. Circuits Syst. II: Express Briefs*, accepted for publication.

[28] S. Haykin, *Communication Systems*, 4th ed. New York, NY, USA: Wiley, 2000.

[29] X. Gao, E. Klumperink, M. Bohsali, and B. Nauta, "A 2.2 GHz 7.6 mW sub-sampling PLL with -126 dBc/Hz in-band phase noise and 0.15 psrms jitter in 0.18 μm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2009.



Jiayoon Zhiyu Ru (M'09) received the Bachelor degree from Southeast University, Nanjing, China, in 2002, the Master degree from Lund University, Lund, Sweden, in 2004, and the Ph.D. degree from the University of Twente, Enschede, The Netherlands, in 2009.

In 2004, he did the Master project with Ericsson in Lund, working on digital TV receiver. From 2005 to 2009, he did Ph.D. research in the IC-Design group of Twente, working on software-defined radios. From 2010 to 2012, he did postdoctoral research in the same group on DTC and digital PLL. In 2011, he was a visiting scientist at MIT. From 2012 to 2014, he was with MediaTek in Boston. He is currently with Qualcomm in San Diego, CA, USA.

Claudia Palattella (S'13) received the M.Sc. degree (*cum laude*) in electrical engineering from the Politecnico di Milano, Milan, Italy, in 2012. She is currently working towards the Ph.D. degree at the Integrated Circuit Design group at University of Twente, Enschede, The Netherlands. Her research interests include frequency synthesis, oscillators, and discrete time systems.

Paul Geraedts (S'07–M'10) was born in Deventer, The Netherlands, in 1979. He received the M.Sc. degree in electrical engineering from the University of Twente, The Netherlands, in 2005.

He is currently working on analog-to-digital converters at Teledyne DALSA, Enschede, The Netherlands.

Eric A. M. Klumperink (M'98–SM'06) was born on April 4, 1960, in Lichtenvoorde, The Netherlands. He received the B.Sc. degree from HTS, Enschede, The Netherlands, in 1982. After a short period in industry, he joined the University of Twente in 1984, participating in analog CMOS circuit research resulting in several publications and his Ph.D. thesis "Transconductance Based CMOS Circuits" (1997). In 1998, he became an Assistant Professor at the IC-Design Laboratory in Twente and his research focus changed to RF CMOS circuits. In April–August 2001, he extended his RF expertise during a sabbatical at the Ruhr Universitaet in Bochum, Germany. Since 2006, he has been an Associate Professor, teaching Analog and RF IC Electronics courses. He participates in the CTIT Research Institute, guiding Ph.D. and M.Sc. projects related to RF CMOS circuit design with focus on software-defined radio, cognitive radio and beamforming.

Dr. Klumperink served as an Associate Editor for the IEEE Transactions on Circuits and Systems II (2006–2007), IEEE Transactions on Circuits and Systems I (2008–2009), and IEEE Journal of Solid-State Circuits (2011–2013). He is a member of the technical program committees of the IEEE International Solid State Circuits Conference (ISSCC) and IEEE RFIC Symposium. He holds several patents, has authored and co-authored more than 150 internationally refereed journal and conference papers, and is a co-recipient of the ISSCC 2002 and the ISSCC 2009 Van Vessem Outstanding Paper Awards.

Bram Nauta (M'91–SM'03–F'08) was born in 1964 in Hengelo, The Netherlands. In 1987 he received the M.Sc degree (*cum laude*) in electrical engineering from the University of Twente, Enschede, The Netherlands. In 1991 he received the Ph.D. degree from the same university on the subject of analog CMOS filters for very high frequencies.

In 1991 he joined the Mixed-Signal Circuits and Systems Department of Philips Research, Eindhoven, The Netherlands. In 1998 he returned to the University of Twente, as a Full Professor heading the IC Design group. In 2014, he was appointed as a Distinguished Professor at the University of Twente. His current research interest is high-speed analog CMOS circuits, software-defined radio, cognitive radio, and beamforming.

Dr. Nauta served as the Editor-in-Chief (2007–2010) of the IEEE Journal of Solid-State Circuits (JSSC), and was the 2013 program chair of the IEEE International Solid-State Circuits Conference (ISSCC). He served as an Associate Editor of IEEE Transactions on Circuits and Systems II (1997–1999), and of JSSC (2001–2006). He was in the Technical Program Committee of the Symposium on VLSI circuits (2009–2013) and is on the steering committee and program committee of the European Solid-State Circuits Conference (ESSCIRC). He is also a member of the ISSCC Executive Committee. He served as Distinguished Lecturer of the IEEE, and is an elected member of the IEEE-SSCS AdCom. He was a co-recipient of the ISSCC 2002 and 2009 Van Vessem Outstanding Paper Awards and in 2014 he received the Simon Stevin Meester Award, the largest Dutch national prize for achievements in technical sciences.