# UNIVERSITY OF TWENTE.

## Faculty of Electrical Engineering, Mathematics & Computer Science

# Design and implementation of an Analog-to-Time-to-Digital converter

J.D.A. van den Broek
Master's thesis
October 2012

## Acknowledgement

## Summary

This thesis describes the design and implementation of an analog-to-digital converter (ADC) taking an uncommon two-step approach: a voltage-to-time converter (VTC) converts the analog input signal to a difference in time between two digital transitions. Subsequently, a time-to-digital converter (TDC) quantizes this time difference to yield the digital output code.

The aim of the thesis was to investigate the fundamental advantages of such a topology and to demonstrate them by implementing a proof-of-concept.

Since TDCs can be highly digital structures, this approach was expected to yield an architecture with a mostly digital structure and work flow, ensuring good portability to, and inherent improvement with, newer CMOS technology. Another possible advantage is that in many TDCs, conversion time can be traded for accuracy; such reconfigurability is not common in conventional ADCs.

After a comprehensive study of TDC concepts, VTC concepts and existing analog-to-time-to-digital converters, a novel architecture is proposed. It comprises a free-running ring oscillator and associated digital logic to form the TDC and a start-voltage controlled single slope converter to form the VTC. A reference sampling mechanism is used for insensitivity to most low-frequency variations and noise sources.

The VTC consists of multiple channels that make use of the TDC in an interleaved fashion, distributing the power consumption of the TDC over multiple conversions, which is beneficial to the system performance. The multiple-channel VTC can also operate as one channel with higher accuracy, demonstrating the reconfigurability aspect of the analog-to-time-to-digital converter.

A proof of concept was largely implemented on transistor level in 140 nm CMOS. The implemented circuit is indeed highly digital and the analog parts implemented so far are carefully picked for technology scalability.

Although the final figure-of-merit (FoM) of the system is about one order of magnitude from state-of-the-art, most aspects of the performance are dominated by the digital circuitry. Therefore, the architecture is expected to improve rapidly when the concept is ported to a newer CMOS technology.

# Contents

# List of Acronyms

| | |
|---|---|
| ADC | Analog-to-digital converter |
| CMOS | Complementary metal-oxide-semiconductor |
| DNL | Differential nonlinearity |
| DR | Dynamic range |
| ENOB | Effective number of bits |
| FoM | Figure-of-merit |
| GRO | Gated ring oscillator |
| INL | Integral nonlinearity |
| LSB | Least-significant bit |
| PLL | Phase-locked loop |
| PVT | Process, voltage and temperature |
| PWM | Pulse-width modulation |
| SAR | Successive approximation (register) |
| SINAD | See SNDR |
| SNDR | Signal-to-noise-and-distortion ratio |
| SNR | Signal-to-noise ratio |
| SSP | Single-shot precision |
| TA | Time amplifier |
| VCDL | Voltage-controlled delay line |
| VCO | Voltage-controlled oscillator |
| VDL | Vernier delay line |
| VTC | Voltage-to-time converter |

# 1 Introduction

With the improvement of CMOS technology, transistors become ever smaller and faster, while supply voltages are reduced. Whereas these developments allow for faster, smaller and more power-efficient digital circuitry, they make analog and mixed-signal circuit design increasingly challenging.

For instance, creating matched and ratioed voltages and currents with low noise, an essential task in data conversion, becomes more difficult with smaller components and little voltage headroom. Meanwhile, the timing properties of transistors, such as rise and fall times and transition frequencies, benefit from technology scaling. This has led some IC-designers to state that timing resolution has become superior to voltage resolution in modern processes [1].

So-called time-to-digital converters (TDC's) utilize this time-domain resolution to quantize the time interval between two events, usually two digital transitions. TDC's are often highly digital circuits. They have been around for some time, originating in nuclear research during the tube era [2, 3] and find other very specific applications today, for example in all-digital phase-locked loops [4–11], on-chip jitter measurement [12] and laser-range-finding [13].

If a time-to-digital converter were preceded by an analog-to-time converter, the result could be an analog-to-digital converter (ADC) with some interesting prospects, as detailed in section 1.2. Assuming an ADC with a voltage input, the analog-to-time converter will be referred to as voltage-to-time converter (VTC) in this work. This thesis describes the analysis and design of such an 'analog-to-time-to-digital' converter topology.

## 1.1 Research scope

A block level overview of the ADC topology of interest is given in figure 1. A 'reference' digital transition is fed through a block in which it experiences a delay $t_d$ that is linearly dependent on the input signal. This block will be referred to as the VTC. In the next block the interval between the reference transition and the delayed transition is digitized. This block is referred to as the TDC.
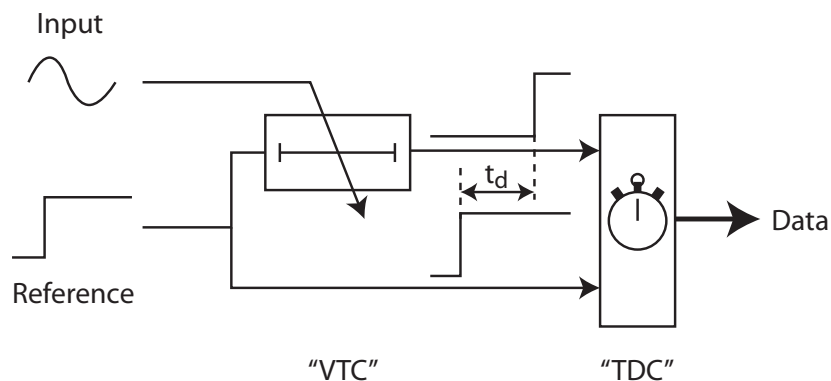


**Fig. 1:** Overview of the analog-to-time-to-digital converter concept.

To limit the scope of this work, only architectures are considered in which the VTC and TDC are distinctly separate blocks. This excludes other time based ADC topologies in which the input voltage is used directly as a control voltage for the time base, such as voltage-controlled oscillator (VCO)-based ADCs [14] and voltage-controlled delay-line (VCDL)-based ADCs [15–17]. This is done for two reasons: First, the input V-to-I converter in such ADCs is hard to linearize beyond a few bits without much analog effort or digital correction. Second, they often have an integrating input (from voltage to phase) instead of a sampling input, which makes them unusable beyond Nyquist.

Strictly spoken, a clocked counter may already be referred to as a TDC. To exclude this trivial solution form this research, only TDC architectures are considered that achieve time resolutions higher than that of a simple counter,

i.e. in the order of one gate delay or less.

## 1.2  Motivation

There are several interesting prospects for the proposed type of time-based ADC.

As detailed in the next section, TDC's are generally highly digital structures. As a result, many aspects of their performance benefit from technology scaling: speed, time resolution, power consumption and area. Any feature of the overall ADC that is dominated by the TDC will therefore inherently improve with newer CMOS technology. Besides this, highly digital structures are often easily implemented and ported to newer technologies.

Another interesting aspect is the following: given that the TDC can achieve a certain time resolution (e.g. a least-significant bit (LSB) represents 100 picoseconds), there exists an interesting trade-off: if more time is available for the conversion, more levels can be calculated. Such reconfigurability is much less trivial in conventional ADC's.

In short, the analog-to-time-to-digital converter may prove useful in the following areas:

- Inherent improvement with technology,
- Ease of implementation,
- Chip area consumption,
- Reconfigurability,
- Power consumption.

The aim of this thesis is to **design and implement an analog-to-time-to-digital converter and explore the aforementioned advantages in the process.**

## 1.3  Thesis outline

This thesis opens with a chapter on exploratory research, including a thorough study of the many existing TDC techniques. It proceeds with possible architectures for the VTC. Finally, it describes examples of existing analog-to-time-to-ditigal converters.

Armed with this knowledge, the next section describes how a novel system level architecture was derived by reasoning, analysis and preliminary simulations.

A section on implementation follows, systematically deriving transistor-level implementations for all the blocks of the proposed architecture.

The results chapter reveals several performance aspects of the VTC and the TDC, as well as simulation results of the overall system.

Finally, some conclusions, outlooks and proposals for further research conclude this thesis.

## 2 Exploratory research on building blocks and existing converters

This section provides an overview of basic concepts for time-to-digital conversion, voltage-to-time conversion and time-based analog-to-digital conversion, largely borrowed from existing work. It is meant to provide basic understanding and for qualitative comparison of possible solutions. The actual equations that govern the performance of the different topologies are presented in the next sections, when they become relevant to the system architecture and design.

### 2.1 Time-to-digital converters

Several existing TDC concepts are discussed in this section. It will become apparent that many ADC techniques have their direct counterpart in TDCs. As a result, most performance metrics of ADCs, as detailed in appendix A, can be directly applied to TDCs.

To exclude the trivial solution of using a clocked counter, only TDCs are discussed that achieve time resolutions in the order of a gate delay or less.

All TDC techniques make use of delay lines and/or oscillators, which are often placed in a delay-locked loop or phase-locked loop, respectively, to fix part of their behavior to a reference clock. For the sake of clarity, these surrounding loops have been left out of the illustrations and discussions.

Many TDCs output a combination of binary code and (pseudo-)thermometer code. Again, to prevent going into too much detail, the required decoding logic was left out of the illustrations and discussions, as well as any control, timing, correction and calibration circuitry.

#### 2.1.1 Flash TDC

The flash TDC [18–20] is perhaps the most basic TDC within the scope of this work; an overview is shown in figure 2. Its name stems from the analogy with a flash ADC, which typically uses a resistor ladder to create uniformly distributed levels in the voltage domain; a flash TDC uses a delay line to achieve the same in the time domain. This delay line can be implemented in a variety of ways, such as single-ended or differential CMOS inverters, buffers, et cetera.
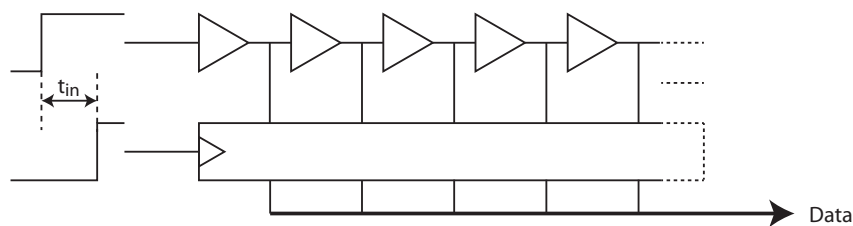


**Fig. 2:** Simplified illustration of a flash TDC.

**Mechanism:** The leading edge is fed into the delay line. During its propagation, the lagging edge arrives (after $t_{in}$ in figure 2), which is used to take a 'snapshot' of the state of the delay line, using latches or flip-flops. The number of delay elements that has toggled by the time the lagging edge arrives, is a linear measure for the time difference between the leading and lagging edge.

**Advantages:**

- The structure is very simple and inherently monotonic, provided that the flip-flops or latches do not show extraordinary amounts of offset.
- Sampling rate can be traded for dynamic range by picking a different length of delay line.

**Limitations:**

- The time resolution of this type of TDC is limited to the propagation delay of one element.

- An upper limit exists for the length of the delay line: for one, because a long delay line consumes a lot of area, but more importantly, the mismatch between the delay elements introduces an accumulating uncertainty in the propagating edge. The latter effect impairs the linearity of the converter. Interestingly, the effect of mismatch in the delay elements is generally an order of magnitude larger than that of thermal noise, as will be demonstrated in section 4.1.1. So unless the mismatch is conquered, this type of TDC is limited by linearity issues rather than thermal noise effects.

- The power consumption is at least the consumption of one toggling delay element and one flip-flop for each level to be calculated.

- The sampling rate is limited by the total delay of the delay line, although pipelining can be used to have multiple transitions in the delay line at the same time [21].

### 2.1.2 Ring delay line TDC

The achievable dynamic range in a flash TDC is limited by impractically long delay lines and by the accumulating effect of mismatch on the propagating edge. Both issues can be conquered by rolling the TDC into a ring, as shown in figure 3. The example shows a 5-stage ring.
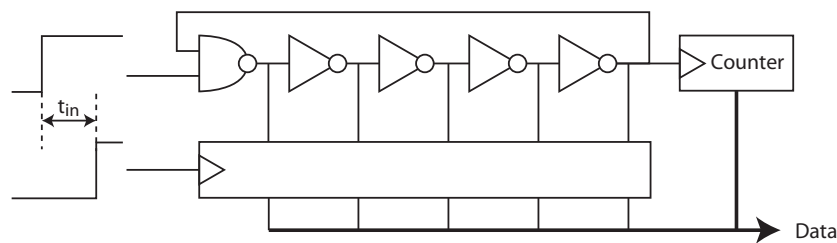


**Fig. 3:** Simplified illustration of a flash ring TDC. The delay elements are now explicitly inverting to form an oscillator.

**Mechanism:** The rising edge triggers a ring oscillator. A counter keeps track of the number of oscillations. The lagging edge takes a snapshot of the state of the ring, which is combined with the state of the counter to provide an output value.

**Advantages:**

- This type of TDC occupies very little area.

- The mismatch of the delay elements translates to a small cyclic linearity error, as shown in figure 4, instead of accumulating like in a flash TDC, since the same delay elements are used cyclically.

**Limitations:**

- Like in a regular flash TDC, the time resolution is still limited to a single propagation delay. Also, power consumption is still limited by the toggling of one delay element for each level. Less flip-flops are required, but a counter is added.

- A ring with a clean start-up behavior is required [22].

- Care has to be taken to load all nodes equally to prevent excessive DNL errors.

- The ring cannot be made extremely short, because the counter has a limited clock frequency.

- If all of these practical issues are covered and the cyclic nonlinearity is sufficiently small, the performance becomes limited by the accumulating effects of thermal noise on the propagating edge. This means that sampling rate can be traded for dynamic range in a far greater range than in the regular flash TDC.
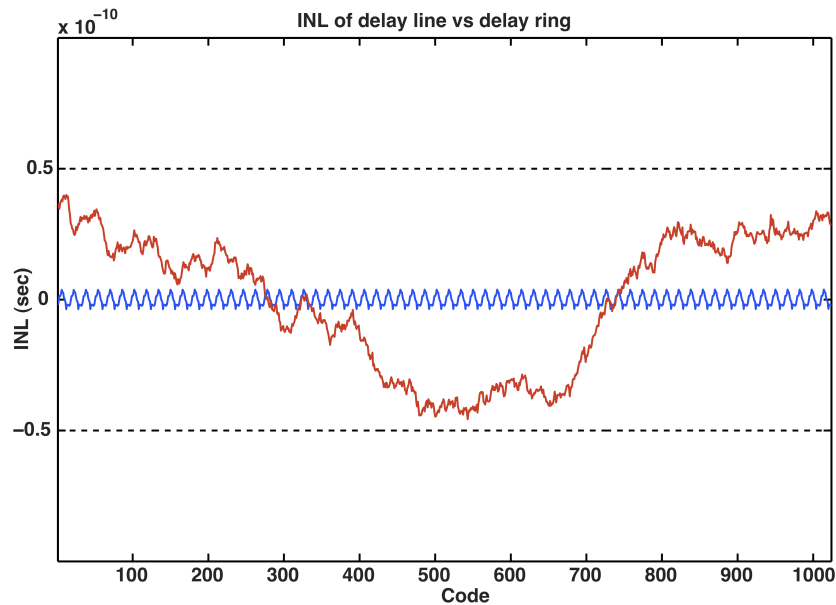
**Fig. 4:** Typical INL profile of a 10-bit TDC (behavioral simulation), when elements are used with a nominal delay of Td = 50ps and a mismatch of $\sigma_{Td} = 3\%$, which are realistic values for small inverters in 140 nm CMOS. The red line represents a straight flash TDC, with the accumulating effect of mismatch clearly visible. The blue line represents a flash ring TDC of 16 stages, in which the mismatch results in a much less severe cyclic INL pattern. The black dashed lines indicate +/- 1LSB.

### 2.1.3   Vernier delay line TDC

The time resolution of flash TDCs is limited to the propagation delay of one element, even if the matching of the elements would allow much better. Some applications, such as digital PLLs, even demand a higher resolution TDC to meet their phase noise specifications. This issue is addressed by the Vernier delay line (VDL) TDC [19]. Its name stems from the similarity with Vernier scales such as those found on calipers. An impression is shown in figure 5.
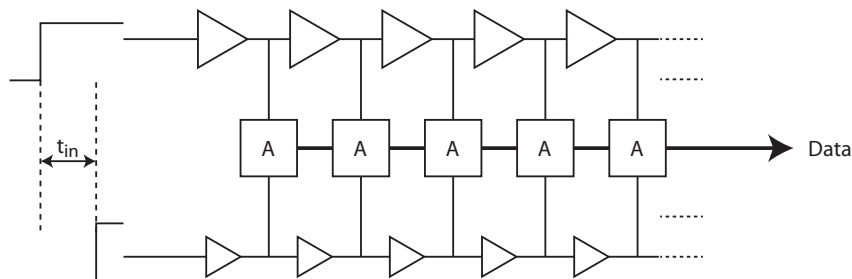


**Fig. 5:** Simplified illustration of a vernier delay line TDC. The smaller delay elements have less delay ('faster'). The 'A' blocks are arbiters, that determine which of two transitions arrives first.

**Mechanism:** The VDL TDC makes use of two parallel delay lines, one slightly faster than the other. The leading edge is sent into the slower delay line, the lagging edge into the faster one. So-called arbiters determine which signal arrives first at each pair of nodes. The point where the lagging edge overtakes the leading edge is a linear measure for the initial time difference.

**Advantages:**

- Compared to a flash TDC, the structure is still relatively simple, but the time resolution is now limited to the difference between a fast and slow element, which can be much smaller than the unit delay itself. In

practice, resolutions of 5 to 10 times below a unit delay can be achieved [19] or even finer resolutions when the elements are calibrated [12].

**Limitations:**

- More so than in the flash TDC, matching between the delay elements limits both the resolution and the dynamic range, as the relative mismatch in a small difference between delay elements is much larger than the mismatch of one delay itself. Care needs to be taken to keep the structure monotonic, and to keep the delay line short enough for sufficient linearity.

- The power consumption is generally larger than in a flash TDC, since twice as many delay elements need to make a transition for the same number of bits. Furthermore, the more stringent requirements on matching of delays may require larger transistors compared to a flash TDC, also resulting in a higher power consumption.

- Resolving the small time differences by the arbiters usually takes long compared to the time steps themselves. Therefore VDL TDCs have a large dead-time to resolve their outputs, recover and prepare for the next conversion.

### 2.1.4  Vernier ring TDC

Like the flash TDC, VDL TDCs are not very scalable; to achieve a high dynamic range, long delay lines are required and matching starts to impair the linearity. But like the flash TDC, a VDL TDC can be rolled into a ring [23, 24]. The concept is shown in figure 6 for a number of five stages. Interestingly, an early TDC from the 1950s was actually a one-stage Vernier ring TDC made of tubes, called a 'Vernier chronotron' [2].
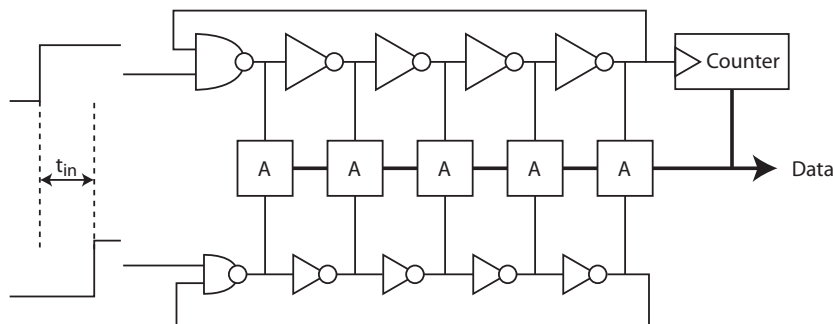


**Fig. 6:** Simplified illustration of a Vernier ring TDC. The ring oscillator with smaller elements has a higher frequency.

**Mechanism:** A Vernier ring consists of two ring oscillators with an equal number of stages, but slightly different delays per stage; in this case five stages are shown. The leading edge triggers the slower oscillator, the lagging edge triggers the faster oscillator. In between these events, several oscillations of the slower oscillator may have occurred, which are counted to form the coarse data. After the lagging edge arrives, the Vernier principle starts working until coincidence is detected, yielding the fine data.

**Advantages:**

- This structure is relatively simple, and able to reach sub-gate-delay resolution.

- It features a large dynamic range in a small area.

- The faster ring is only active from the time the lagging edge arrives until it overtakes the leading edge. This is beneficial to the power consumption.

- Like in the flash ring TDC, mismatch errors now have a cyclic contribution, and therefore less impact on the linearity of the TDC.

**Limitations:**

- Even more so than in a flash ring TDC, equal loading of the nodes and low offsets of the arbiters are required to maintain monotonicity and low DNL.

- If proper measures are taken against the effects of mismatch, thermal noise becomes dominant in the achievable dynamic range of this TDC.

### 2.1.5 Pulse shrinking delay line TDC

The pulse-shrinking delay line TDC [25] is functionally very similar to the VDL TDC. An overview is shown in figure 7.
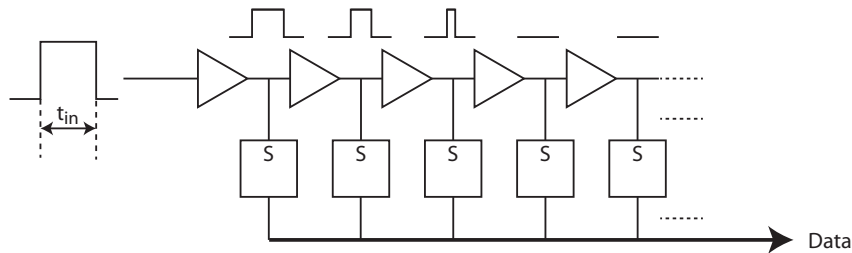


**Fig. 7:** Simplified illustration of a pulse-shrinking delay line TDC. The blocks with 'S'-inputs are set-flip-flops (reset not shown).

**Mechanism:** This TDC consists of a delay line that has different propagation speeds for rising and falling edges. Suppose rising edges travel slower than falling edges. In this case, the leading edge is sent into the delay line as a rising edge, the lagging edge is sent into the delay line as a falling one. A traveling pulse results, that sets SR-latches on its way. As the lagging edge catches up with the leading edge, the travelling pulse vanishes and fails to toggle subsequent latches. The position where this occurs, is a linear measure for the input time difference.

**Advantages:**

- Compared to the VDL TDC, this structure requires only one delay line.
- The delay line has automatically returned to its initial state after the conversion is over; it requires no reset.

**Limitations:**

- This topology requires delay elements with asymmetrical propagation delays. The difference in propagation delay between rising and falling edges must be well-controlled and well-matched between stages to obtain sufficient linearity.

### 2.1.6 Coarse-fine TDC

All aforementioned TDCs are fundamentally limited in terms of power consumption, because they require one or two delay elements to toggle for each level to be calculated. This can be solved by coarse-fine conversion, an example is discussed in [9]. An simplified version is shown in figure 8.

**Mechanism:** This TDC uses a low-power ring oscillator and counter to coarsely digitize the time interval. After the lagging edge has arrived, the counter is read out, and the remainder of that coarse oscillator period is sent to a fine TDC, depicted here as a simple flash TDC. If the coarse-to-fine TDC gain is known, the digitized remainder of the oscillation period can be used to calculate the fine bits that are appended to the coarse data. The fine converter can be any one of the aforementioned concepts. In [9], four interleaved flash TDCs are used, but a VDL TDC may also be used.

**Advantages:**

- The coarse oscillator can be a relatively slow, low-power oscillator, provided that its jitter will not dominate the fine conversion.
- The fine converter is only active during one period of the coarse converter, drastically reducing the power required for one conversion.

**Limitations:**
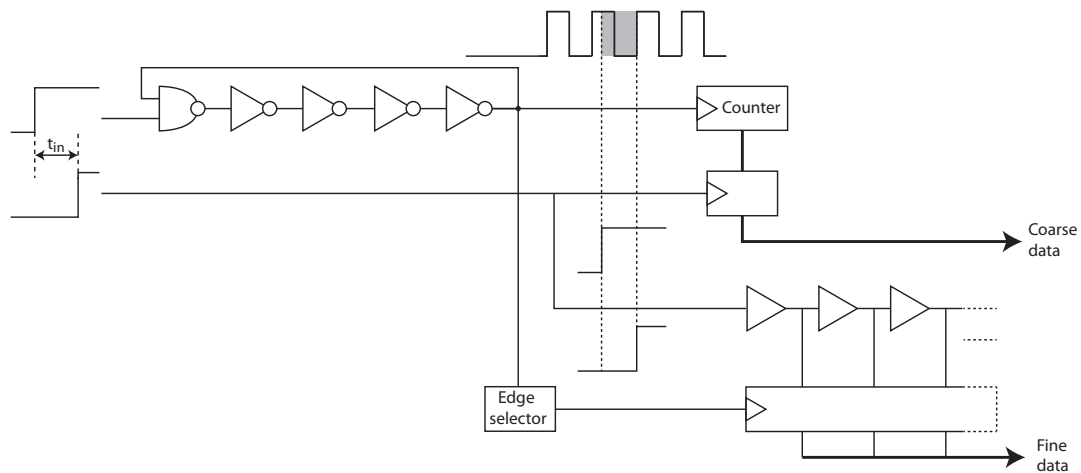
**Fig. 8:** Simplified illustration of a coarse-fine TDC. A ring oscillator and counter are used for coarse quantization, a flash TDC is used for fine quantization of the remainder of the last oscillation period (gray area).

- In this structure, especially the coarse-to-fine converter gain needs to be either measured or established. In [9], this is measured by mechanisms that rely heavily on the chaotic locking behavior of the PLL in which the TDC is applied. This makes the structure less attractive for use in an ADC.

### 2.1.7  Time-amplifying TDC

A completely different coarse-fine approach makes use of time amplification [26]. An overview is shown in figure 9.

Linear amplification of a time interval can be done using 'time amplifiers' (TAs). Such TAs are based on latches; a latch shows an exponentially increasing decision time when the input time difference becomes very small (near metastability). Combining the characteristics of two asymmetrical latches yields a region in which linear time amplification is obtained [26, 27].
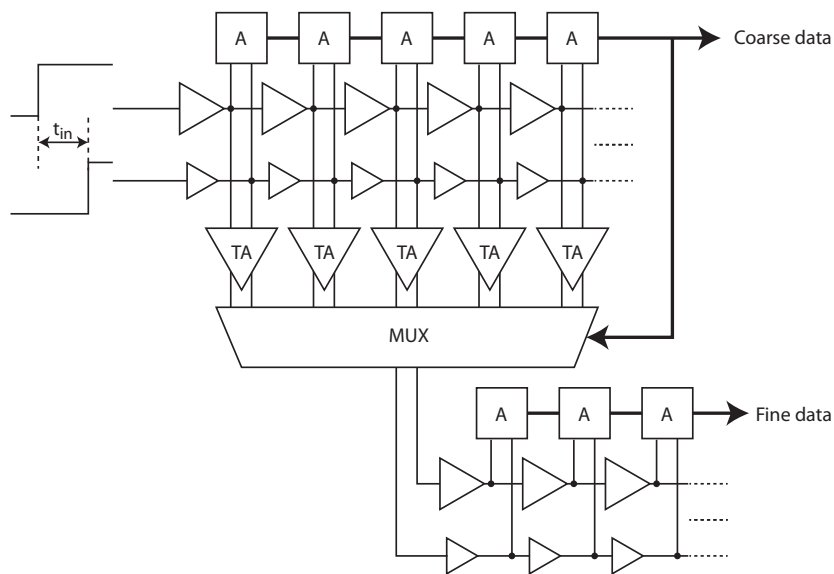


**Fig. 9:** Simplified illustration of a coarse-fine TDC using time amplification.

**Mechanism:** The time interval is first digitized using a coarse VDL TDC. Meanwhile, all possible 'time residues'

are amplified using time amplifiers, since time cannot be stored for later use. After the coarse value is determined, the correct 'time residue' can be selected using a multiplexer, for fine quantization using a second VDL TDC.

**Advantages**

- Because of the amplification, very fine time resolutions can be achieved using VDL TDCs of modest resolution.

- For $N + M$ bits, only VDL TDCs of $2^N$ and $2^M$ delay elements are needed instead of $2^{N+M}$. Because of these two short TDCs, the accumulating effect of mismatch is less severe, so matching requirements are relaxed. Relaxed matching requirements and shorter delay lines are beneficial to area and power consumption.

**Limitations:**

- The structure presents several design challenges: Time amplifiers suffer from gain, offset and linearity problems much like voltage amplifiers. Although the authors of [26] leverage very clever techniques to monitor and correct these issues, these are not easily adopted.

- This type of TDC requires a large dead time, compared to its full-scale input time. This is necessary for the time-amplifiers to function and the outputs to stabilize.

### 2.1.8  Successive approximation TDC

In successive approximation (SAR) TDCs, a binary search is executed to align the leading and lagging edge. Figure 10 shows a simplified example.
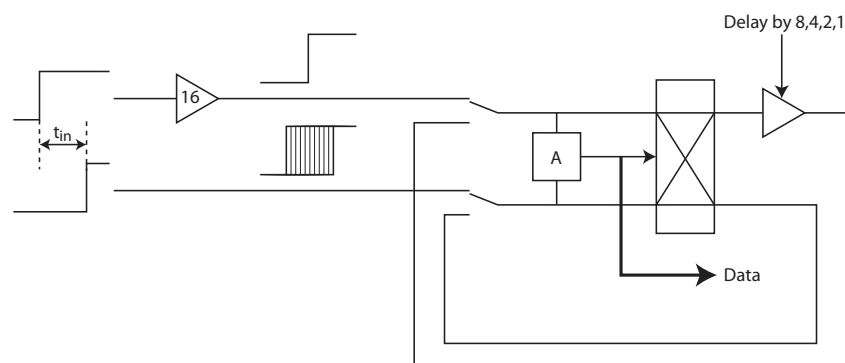


**Fig. 10:** Simplified illustration of a successive approximation TDC. The depicted loop calculates 5 bits.

**Mechanism:** The leading edge is delayed by half the full-scale time. An arbiter determines which of the two edges now arrives first. The output of the arbiter serves as the most significant bit. The early edge is now delayed by a quarter of the full-scale time and the process repeats. This process is repeated until the achievable number of bits is reached.

In the form of figure 10, the structure operates in a loop. However, this requires a programmable, binary weighted delay, along with associated control logic. The loop can also be unrolled to form a less complicated structure, at the expense of area [28]

**Advantages:**

- SAR TDCs require at least as much total delay to perform their function as flash TDCs (e.g. an ideal 10-bit SAR TDC still requires about 1024 delays in total), however they are lumped into a delay of 512, one of 256, one of 128 et cetera. This can be a practical advantage; details can be found in appendix C.

- The SAR structure can achieve sub-gate delay resolution when the difference in delay between the paths is below one gate delay (e.g. one path introduces 1 extra unit delay, the other introduces 1.5 unit delays). In other words, the finest SAR stages can be Vernier stages.

**Limitations:**

- In both the cyclic and the unrolled version, the edges need to be 'held up' while the arbiters decide which edge should be delayed for the next decision. To avoid metastability, these 'buffer delays' should be quite long. Any mismatch in these delays will add to the nonlinearity of the system, so some form of calibration quickly becomes unavoidable.

### 2.1.9   Oversampling TDC

Certain TDC topologies show noise-shaping behavior, and can therefore be used as an oversampling TDC. An example is the gated ring oscillator (GRO) [29], as depicted in figure 11.
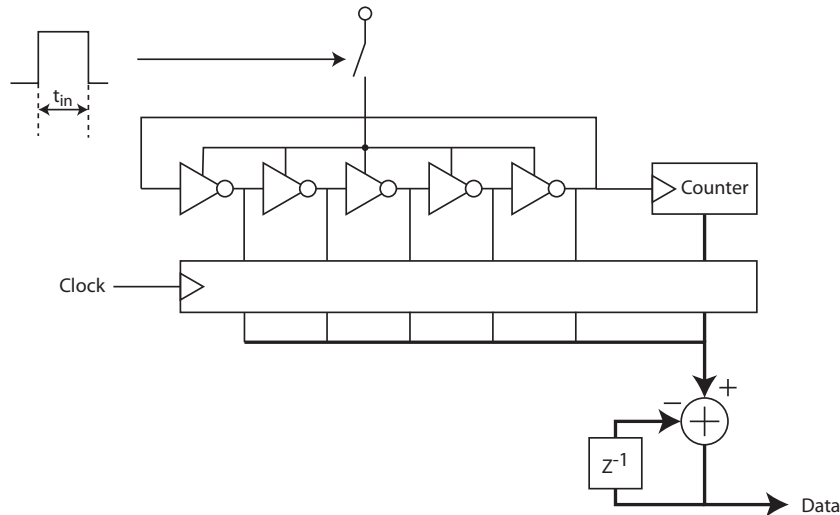


**Fig. 11:** Simplified illustration of a gated ring oscillator TDC.

**Mechanism:** The leading edge starts a ring oscillator. A counter starts registering the amount of periods the oscillator makes. At the arrival of the lagging edge, the ring oscillator is stopped and the internal state of the oscillator is registered. The state of the oscillator from the previous sample is subtracted, leaving only the phase increase during the current sample.

This method in itself does not yet provide noise shaping. However, when the ring oscillator is stopped by cutting its power, some charge remains on its internal nodes. This charge represents some excess phase, that was too small to increase the state of the oscillator by one. In other words, this represents a quantization error. The next time the oscillator is started, it starts a little ahead of the measured phase, effectively subtracting this quantization error. This mechanism provides first-order noise shaping. Besides this, the inherent element rotation also provides first order shaping of the delay mismatches between the ring oscillator stages.

**Advantages:**

- If the technology and the application permits oversampling of the signal, a higher resolution can be obtained using a GRO TDC.

**Limitations:**

- For correct preservation of the charge, some analog design effort is required to prevent leakage and charge-injection from degrading the noise-shaping behavior.

**Final remarks:** The GRO topology is a special case of a ring oscillator switching between two frequencies, one of which is zero. Completely stopping the oscillator has some drawbacks, such as leakage of the charge that represents the quantization error and start-up effects of the oscillator. These drawbacks are addressed by the differential switched ring oscillator (SRO) topology given in [30]. It toggles two oscillators between a high and a low frequency rather than completely disabling them. Two oscillators are used to obtain an overall differential architecture.

Even higher order noise shaping TDCs have already been successfully implemented, for example a 1-1-1 MASH topology, providing third-order noise shaping [31].

### 2.1.10 Delay interpolation

One major topic that was not treated in any of the aforementioned concepts is delay interpolation: If sub-gate delay resolution is desired, but switching to a Vernier concept is too tedious, interpolation of delay elements can be applied. Examples are using multiple flash TDCs in parallel [8], resistive interpolation [32], or read-out using interpolating flip-flops [5].

### 2.1.11 Quantitative comparison of TDCs

Table 1 gives an impression of the large variety of specifications achievable with the different TDC topologies. Some performance measures are specific to TDCs, others General trends are hard to distinguish, but key parameters such as time resolution and figure of merit seem to benefit from technology scaling. Most concepts show a power consumption in the order of several mW.

Some authors have started using the Walden figure-of-merit (FoM) to compare TDCs, but the varying amount of information available on the different TDCs makes it hard to do the competition justice. For instance, the FoMs of [32] and [26] are calculated to be 309 and 994 fJ/conversion-step, respectively, by the author of [8], but are found to be 190 and 2340 fJ/conversion-step, respectively, by the author of [28]. One issue that prevents an honest comparison is that many TDC topologies show a signal-dependent power consumption. Another is that the varying inclusion of digital blocks in the power consumption. To prevent speculation based on the varying amount of data available from each publication, only FoMs are shown as calculated by their own respective authors. However, the few FoMs that are available definitely show potential for use in an ADC.

A remarkably low FoM is that of [8]. Although the topology seems indeed very power-efficient, the FoM is a dubious one: It uses both the highest effective number of bits (ENOB) achievable (13.3), and the highest sampling rate achievable (40 MS/s) in the calculation of the FoM. However, the topology cannot achieve these at the same time: digitizing the full-scale time interval of 90112 ps is physically only possible at 11 MHz or less. Therefore it seems that the actual FoM should be about a factor four higher, at 24 fJ/conversion-step, which is still remarkably efficient, thanks to the coarse-fine conversion mechanism.

### 2.1.12 Conclusion on TDC architectures

From this study of many TDC architectures published to date, some general conclusions can be drawn. First, choosing a Vernier topology has little advantages besides sub-gate delay time resolution, and poses many additional challenges, so is best avoided when the application does not strictly require sub-gate-delay resolution. This is reinforced by the solution from [8], which achieves a very good figure-of-merit with a relatively easy to implement solution. The latter also demonstrates that coarse-fine conversion can be key in achieving a good FoM. Finally, the comparison table shows that TDCs benefit from CMOS technology scaling and state-of-the-art topologies achieve FoMs in a range that is attractive for their use in ADCs.

## 2.2 Voltage-to-time converters

At the core of any common form of voltage-to-time conversion lies a current-source / capacitance combination, generating a linear ramp, and some form of threshold detector. The difference lies in which variable is controlled by the input voltage. Theoretically, the options are to control the current, the capacitance, the threshold voltage or the start voltage of the ramp.

**Tab. 1:** Comparison of existing time-to-ditigal converters

| Source | [12] | [27] | [26] | [33] | [34] | [29] | [31] | [32] |
|---|---|---|---|---|---|---|---|---|
| Technology (nm) | 90 | 180 | 90 | 90 | 500 | 130 | 130 | 90 |
| Topology | VDL + mismatch compensation | Cascaded time amplifier | Coarse-fine VDL + time amplification | Multi-channel GRO | Flash + VDL | Multi-path GRO | 1-1-1 MASH | Passive interpolation |
| Power (mW) | - | - | 3 | 4.8 | | 2.2 - 21 | 1.7 | 3.6 |
| Input range (ps) | - | 1300 (60) | 640 | 12500 | >370 | 12288 | 100000 | 601.6 |
| No. of bits | - | 7 (6) | 9 | - | | 11 | - | 7 |
| Resolution (ps) | 0.88 | 10.2 (1.0) | 1.25 | 17 | 30 | 6 (1) | N/A | 4.7 |
| Single shot precision (LSB) | 0.42 | 1 (8) | 0.6 | 0.08 | - | - | N/A | 0.7 |
| Sample rate (MS/s) | 1000 | 10 (40) | 10 | 40 | - | 50 | 5 | 180 |
| Bandwidth (MHz) | 500 | | 5 | 20 | - | 1 | 0.1 | 90 |
| FoM (pJ/conv) | - | - | - | - | - | 0.2 | - | - |
| Area (mm2) | 0.45 | 0.52 | 0.6 | 0.3 | 0.53 | 0.04 | 0.11 | 0.02 |

| Source | [25] | [35] | [36] | [30] | [28] | [8] | [23] | [13] |
|---|---|---|---|---|---|---|---|---|
| Technology (nm) | FPGA | 350 | 130 | 90 | 65 | 40 | 130 | 800 |
| Topology | Pulse-shrinking | Cyclic SAR | SAR | Switched ringosc | SAR | Coarse-fine | Vernier ring | Counter + interpolation |
| Power (mW) | - | 33 | 1 | 2 | 9.6 | 0.6 - 1.8 | 7.5 | 350 |
| Input range (ps) | 11500 | 3276800000 | 1984 | 2000 - 840000 | 10000 | 90112 | 32000 | 2500000 |
| No. of bits | 8 | 28 | 6 | - | 10 | 14 | 12 | >16 |
| Resolution (ps) | 42 | 1.22 | 31 | - | - | 5.5 | 8 | 32 |
| Single shot precision (LSB) | 1.33 | 2.62 | 0.28 | - | 0.47 - 0.59 | 0.8 | 1 | 0.94 |
| Sample rate (MS/s) | >1 | ≤ 5 | 500 | 50-750 | 80 | 40 | 15 | <0.1 |
| Bandwidth (MHz) | > 0.5 | ≤ 2.5 | 250 | 1 | 40 | 20 | 7.5 | <0.05 |
| FoM (pJ/conv) | - | - | - | - | 0.23 | 0.006 | - | - |
| Area (mm2) | N/A | 4.45 | 0.15 | 0.02 | 0.11 | 0.01 | 0.26 | 11.9 |

### 2.2.1 Current-controlled VTC

Controlling the current of the VTC requires a linear voltage-to-current converter. There are two commonly used ways to accomplish this: the first is a resistor into the virtual ground node of an active integrator, the other is a MOS-based voltage-controlled current source.

Both techniques usually have integrating inputs, in which the input is not sampled, but continuously converted into the current that is integrated onto a capacitance. This saves the area of a dedicated sampling capacitor. However, since integration corresponds to convolution with a rectangle in the time domain, it yields inherent low-pass filtering at the ADC input [15], meaning that such an input stage cannot be used all the way up to Nyquist, and certainly not beyond (e.g. for IF-sampling). However, the same mechanism does provide useful anti-aliasing.

**Active integrator VTC**

A simplified overview of an active integrator VTC is shown in figure 12. This configuration is typically used for dual-slope conversion [37]: first, the unknown quantity (Vin) is integrated for a fixed time. Next, a known quantity (Vref) is integrated and the time is measured. A major advantage of this structure is that many circuit imperfections cancel, such as nonlinearity of the capacitance. Also, the exact frequency of the time base is unimportant, as the ratio between the known and measured time directly represents the ratio between the input and the reference voltage.
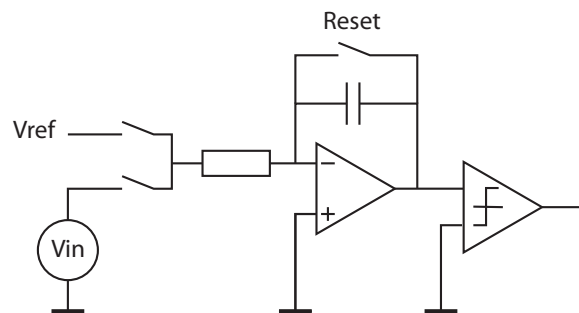


**Fig. 12:** Simplified illustration of an active integrator VTC. Note that for this simplified schematic to be functional as a dual-slope converter, Vref has to have the opposite sign of Vin.

A disadvantage of this structure is that an op amp has to be constructed, which becomes more problematic as CMOS technology scales.

**Voltage-controlled current source VTC**

Figure 13 shows a simple VTC in which the current is regulated by the input voltage, in this case a voltage-controlled delay line (VCDL). This example shows that even in a VTC consisting of multiple stages, a current source, capacitors and threshold detectors can be distinguished, as stated at the beginning of this section.
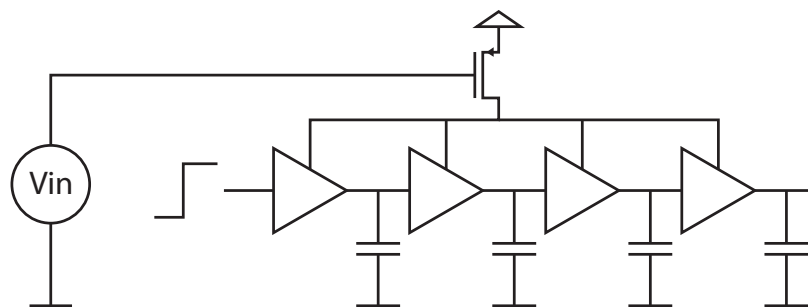


**Fig. 13:** Simplified illustration of a voltage-controlled current source VTC.

Although this type of VTC can be very simple, obtaining a linear control characteristic of the delays is tedious, an

issue well known from VCO-based ADCs. Digital correction is needed to exceed about 6 bits of linearity without much analog effort.

### 2.2.2 Start-voltage controlled VTC

A simplified VTC in which the voltage controls the start voltage of the ramp is shown in figure 14. The start voltage is simply sampled onto a capacitor, after which the capacitor is discharged until the voltage crosses a threshold. A possible advantage is that the sample is taken and the ramp is generated on the same capacitor. The drawback of the depicted implementation is that a current source is needed with a very high output impedance.
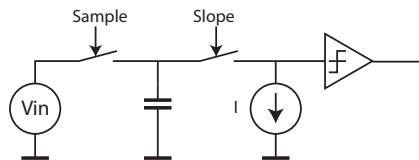


**Fig. 14:** Simplified illustration of a start-voltage-controlled VTC.

### 2.2.3 Threshold-voltage controlled VTC

In figure 15, a simple threshold voltage controlled VTC is depicted. The input voltage is continuously compared with a sawtooth-shaped waveform. The challenges are similar to those in the start-voltage controlled VTC, since a sufficiently linear and low-noise sawtooth generator has to be constructed using a current source and capacitor. If the structure is to be used near Nyquist or beyond, a sampler is also required.
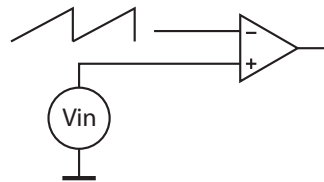


**Fig. 15:** Simplified illustration of a threshold-voltage-controlled VTC.

### 2.2.4 Capacitance-controlled VTC

Figure 16 shows a possible implementation of a VTC in which the capacitance is controlled by the input voltage. The capacitances in this case are MOS capacitances, which can be implemented in various ways in practice. MOS capacitances have a nonlinear control characteristic, resulting in a VTC that can be combined only with TDCs of modest resolution.



**Fig. 16:** Simplified illustration of a capacitance-controlled VTC.

### 2.2.5 Conclusion on VTC topologies

The four theoretically possible VTC topologies were treated conceptually. Start-voltage and threshold-voltage controlled VTCs are the best candidates to achieve good linearity performance with little analog design effort. Current-controlled and capacitance controlled solutions are expected to show modest linearity and are therefore only suitable for use with TDCs of modest resolution. Furthermore, a start-voltage controlled VTC has a sampling input, and can therefore be used up to the Nyquist frequency and possibly for subsampling applications, whereas VTCs with an integrating input provide useful anti-aliasing. A practical advantage of start-voltage controlled VTCs is that the sampling capacitor may be re-used for the slope conversion.

## 2.3 Analog-to-time-to-digital converters

Only few publications on analog-to-time-to-digital conversion with strictly separated VTC and TDC are available as of yet. Three examples will be treated next. For completeness, a fourth ADC is treated that occupies a gray area between the scope of this work and VCO-based ADCs. The performance figures of the four topologies are listed in table 2 for ease of comparison.

**Tab. 2:** Comparison of existing analog-to-time-to-digital converters.

| Reference | [38] | [39] | [40] | [41] |
|---|---|---|---|---|
| Technology (nm) | 180 | 130 | 65 | 65 |
| Sample rate (MS/s) | 10 | 80 | 250 | 1200 |
| Bandwidth (MHz) | 0.5 | 40 | 20 | 600 |
| SFDR / SNR / SNDR (dB) | 66 / ? / 56 | ? / ? / 40.6 | ? / 62 / 60 | 30.1 / ? / 20.4 |
| ENOB (bits) | 9.0 | 6.45 | 9.7 | 3.1 |
| Power (mW) | 4.5 | 6.4 | 10.5 | 2 |
| FoM (fJ/conv-step) | 940 | 920 | 319 | 196 |

### 2.3.1 Start-voltage controlled VTC and GRO TDC

One implementation is given in [38]. The VTC is implemented as a start-voltage controlled VTC: the input is sampled onto a capacitor, which is then discharged through a cascoded current source. The threshold detector is implemented by a single transistor toggling a regenerative latch. This way, the generated edge is already sharp after the first stage of the threshold detector, improving its linearity. The TDC is a gated ring oscillator, and to benefit from its noise shaping, the overall ADC is oversampling by a factor of 20.

The performance figures for this ADC are based on a post-layout simulation.
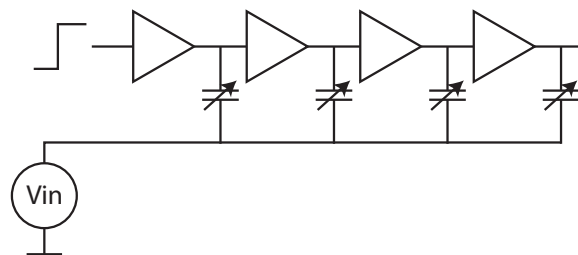
### 2.3.2 Start-voltage controlled VTC and two-step TDC

Another example is [39]. The VTC is again implemented by sampling the input voltage onto a capacitor and discharging it through a triple cascoded current source. The TDC is a two-step TDC, comprising a simple oscillator and counter for coarse quantization, and a multi-path GRO for fine quantization with sub-gate-delay resolution. In this case the oscillator is used at Nyquist, so there is no benefit from noise shaping.

### 2.3.3 Start-voltage controlled VTC and flash TDC in sigma-delta ADC

Another publication worth mentioning is [40], which uses the whole structure of VTC and TDC combined as a quantizer in a sigma-delta ADC. The VTC is an asymmetrical pulse-width modulation (PWM) block, which most closely resembles the 'threshold-voltage controlled VTC' from the previous section. The TDC is a flash TDC that measures the pulse width from the PWM. It also regenerates the discretized pulse to be fed back to the sigma delta loop filter.

### 2.3.4 Voltage-controlled delay-line based ADC

The last work to be mentioned here is that of [41]. The input voltage is sampled differentially and converted to differential control currents for two delay lines. The difference in propagation speed between the two delay lines determines the output code.

Technically, the voltage-to-time and time-to-digital converter are not separate blocks in this architecture: it is a voltage-controlled delay line (VCDL) architecture, placing it outside the scope of this research. However, the work contains useful discussions on noise and mismatch and does show the feasibility of highly digital, time-based architectures.

### 2.3.5 Conclusion on analog-to-time-to-digital topologies

Two ADC topologies that fit the scope of this thesis make use of a start-voltage controlled VTC, the third makes use of a threshold-voltage controlled VTC. This fits the conclusions from the study of VTC architectures. Although the number of studied publications is low and there are large differences in implementation, for the four examples the converter FoM gets better with newer CMOS technology. Also, gated ring oscillators are a popular choice for implementing the TDC.

## 2.4 Summary

The study of existing TDC architectures led to some general conclusions. Key points to keep in mind are that it is possible to keep the structure simple if the application does not strictly require sub-gate-delay performance. Coarse-fine conversion can be key to a good figure of merit of the TDC.

The possibilities for the VTC were explored more conceptually. The start-voltage or threshold-voltage controlled VTC are good candidates to achieve high linearity with limited analog design effort. This is reinforced by the fact that three previous publications that fit the scope of this thesis, also make use of these types of VTCs.

# 3   System-level design of the analog-to-time-to-digital converter

This section outlines and motivates the choices made on system level. A TDC topology is chosen, armed with the knowledge from previous work. Next, a VTC topology is determined, and the interactions between TDC and VTC are discussed. The section concludes with a block diagram of the proposed system.

## 3.1   TDC topology

In choosing a TDC topology, an important realization was that the overall system would be an ADC, with no clear target specification but to make optimal use of the advantages of time-domain quantization. This puts no clear target requirements on the TDC, in contrast to the use of a TDC in a PLL, for example, where the required TDC resolution follows from phase noise requirements, and the required dynamic range follows from the desired range of output frequencies and the range of the feedback divider [5].

So basically all TDC options were open, and a choice needed to be made for a TDC that fit an easily implementable, highly digital, small, reconfigurable and power-efficient ADC topology.

A flash ring TDC turns out to meet most of these criteria. The simplified schematic is repeated in figure 17. It consists of a ring oscillator that can be started by the leading edge, a coarse counter that keeps track of the number of cycles the oscillator has made, and flip-flops to register the state of the oscillator for fine resolution.
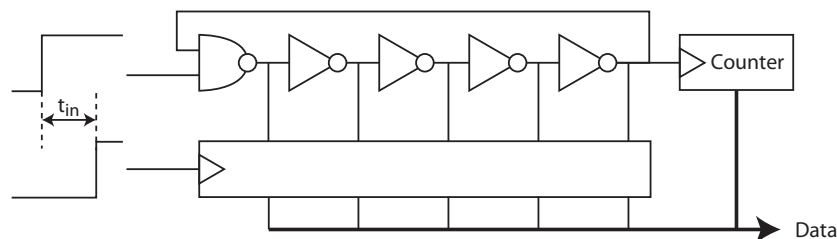


**Fig. 17:** Simplified illustration of a flash ring TDC.

A flash ring TDC is:

- Easily implementable: of all TDC topologies, it poses the most relaxed demands on matching of delay elements. For one, because it does not attempt to achieve sub-gate-delay resolution, and also because mismatch errors are turned into a cyclic pattern. These relaxed demands on matching of delays make this topology very robust to future technology scaling.

- Highly digital: it does not require the design of arbiters, as required in Vernier- or SAR-like topologies.

- Small in area: the ring can be made as short as allowed by the maximum operating frequency of the coarse counter.

- Reconfigurable: By adding an extra bit to the coarse counter and allowing twice as much time for the conversion, the dynamic range is doubled. Since the mismatch error turns into a cyclic pattern of much less than an LSB (as shown in figure 4 in section 2.1.2), this fact can be exploited until the effects of thermal jitter or low-frequency noise become dominant.

However, this type of TDC has one major drawback: Its power consumption is fundamentally limited to the toggling of one delay element for each level. As mentioned previously, one solution is to use coarse-fine quantization, but for this to work, the coarse-to-fine TDC gain has to be either known or well-established. If the TDC is applied in a PLL, an estimate of the coarse-to-fine gain can be made because the chaotic locking behavior of the PLL will eventually hit all the fine codes [8], but in an ADC, the value has to be known regardless of the signal statistics.

This work introduces another way to save power, resulting from the fact that this work is about an ADC, not just a TDC. If the TDC is a ring, that is only read out, but not interrupted for each conversion, it can be used for multiple AD-conversions *in parallel*. In other words, multiple VTC 'channels', operating in parallel, will be

mapped onto one uninterrupted TDC. This way, the power consumption of the TDC can be divided over multiple conversions.

Now, a flash ring TDC that cannot be interrupted for each conversion reduces to nothing but a continuously running ring oscillator, the state of which is sometimes read out. The remaining issue is how to synchronize this oscillator with the incoming time-samples. This can be done in multiple ways:

- register the state of the oscillator at both the leading and lagging edge, and subtract the two values from each other
- force the leading edge to occur when the oscillator is in a known state, so only a snapshot of the state at the lagging edge is needed

The latter option is the most attractive, as it requires only one set of flip-flops to register the state of the oscillator, and less complex digital decoding. The remaining question is how to synchronize the oscillator with the leading edge in practice, but to answer this, first a voltage-to-time topology had to be chosen.

A final note about the TDC ring oscillator is that, to simplify the digital backend, it would be most convenient if the number of stages were a power of two. It will be shown later that with the right oscillator topology, this is indeed possible.

## 3.2   VTC topology

The qualitative comparison of possible VTC topologies in the previous section revealed that an attractive option is to sample the input voltage on a capacitor, then discharge the capacitor using a fixed current. A threshold detector generates the output signal.

Ensuring the linearity of such a VTC is relatively easy. Also, it uses the same capacitor for sampling and generating the slope, possibly saving area. Also, its sampling nature makes it operable up to Nyquist and beyond. All aforementioned publications on analog-to-time-to-digital converters made use of this type of VTC.

The drawback of this topology is that a current source with a very high output impedance is required to keep the current constant while the voltage across the capacitor is dropping. This is hard to implement with small transistors and little voltage headroom. However, the implementation chapter of this work describes a way to overcome this issue. Therefore a sampling, start-voltage controlled VTC was chosen. This fact is used in the remainder of this section.

## 3.3   Integration of TDC and VTC

To integrate TDC and VTC to form an ADC, the issue still had to be addressed of how to force the leading edge to occur when the TDC is in a known state. This can be done in two ways:

- a synchronous way: lock the TDC ring oscillator to the ADC sampling clock using a PLL. Then start the voltage-to-time conversion simply using the sample clock.
- an asynchronous way: let the TDC remain a a free-running ring oscillator. After the input sample is taken, let the TDC indicate when it reaches a known state, and start the voltage-to-time conversion at that instant. The clue here is that the VTC does not necessarily have to start directly after the sample has been taken.

Implementing a PLL requires designing a phase/frequency detector, a loop filter and a charge pump (or, in case of a digital PLL, an additional TDC, digital loop filter and DAC). All of these consume area and power, and require analog design effort. Besides this, most ways of making the oscillator tunable tend to slow it down, degrading the TDC time resolution. Therefore, a free-running ring oscillator will be used for the TDC in this work. So, after the sample has been taken, further timing of the VTC is controlled by the TDC, as illustrated in figure 18.

The drawback of this approach is that the frequency of the ring oscillator will be unknown and varying (due to process, supply and temperature variation and flicker noise). Therefore, additional measures are required to guarantee the TDC gain and accuracy. Both issues were addressed on ADC-level by a reference conversion mechanism, as discussed next.
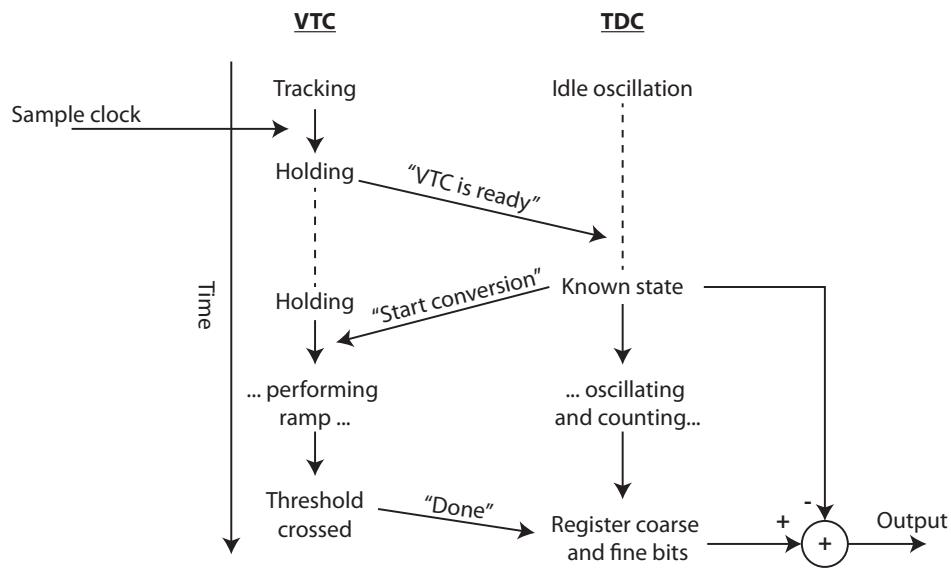
**Fig. 18:** Interaction between the VTC and asynchronous TDC. The ADC sample clock switches the VTC from track to hold mode. Timing of the conversion is governed by the asynchronous TDC.

### 3.3.1   Reference conversion

The final issue to be resolved was how to fix the gain of the ADC over process, voltage and temperature variations. The proposed solution is to periodically sample and convert a reference voltage, next to the input voltage. Such a reference conversion can be used in various ways to either measure or fix the full-scale value of the ADC. If properly applied, it compensates for slow variations in gain in both the VTC and the TDC.

If nothing would be done to use this reference voltage, the waveforms in the VTC would be as depicted in figure 19. At $T_{\text{start}}$, the VTC starts integrating the reference or input voltage with an arbitrary slope. This results in two zero-crossing events, $T_{\text{stop1}}$ and $T_{\text{stop2}}$. $T_{\text{full-scale}}$ represents the full-scale time of the TDC (e.g. for a 10-bit TDC, it represents 1024 unit delays). Note that for clarity, the conversion of $V_{\text{in}}$ and $V_{\text{ref}}$ are depicted as if taking place simultaneously, but this is not necessarily true.
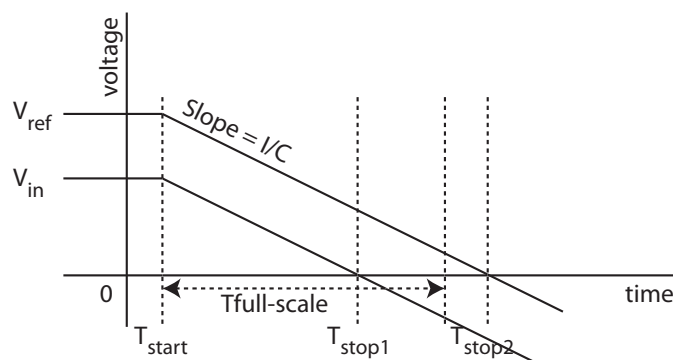


**Fig. 19:** Waveforms and important timing points in a sampling VTC, converting an input and a reference voltage, if no further measures are taken.

There are a few practical ways to obtain useful output from such a VTC. One way is to do nothing more, and just feed $T_{\text{stop1}}$-$T_{\text{start}}$ and $T_{\text{stop2}}$-$T_{\text{start}}$ to a digital divider, as depicted in figure 20.

Another option to fix the ADC gain is to ensure that $T_{\text{stop2}}$-$T_{\text{start}} = T_{\text{full-scale}}$, by adjusting the slope, as depicted in figure 21. This can be done using a first-order locked loop that locks the $T_{\text{stop2}}$ zero-crossing to an edge that is generated when the TDC reaches $T_{\text{full-scale}}$.
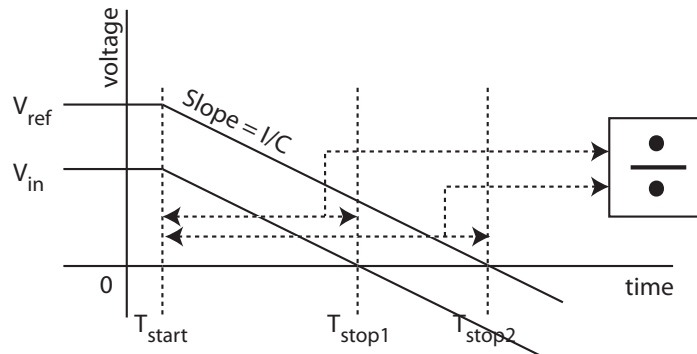
**Fig. 20:** Waveforms and important timing points the VTC, if no further measures are taken and digital division is used to obtain an output value.
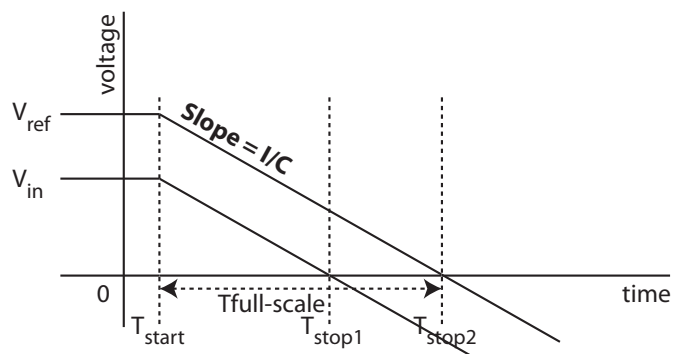


**Fig. 21:** Waveforms and important timing points in the VTC, if the slope is tuned to make $T_{stop2}$-$T_{start}$ equal to $T_{\text{full-scale}}$.

A third option is to do the same, but by adjusting the speed of the oscillator, as shown in figure 22. This can be done in much the same way, again using a first-order loop.
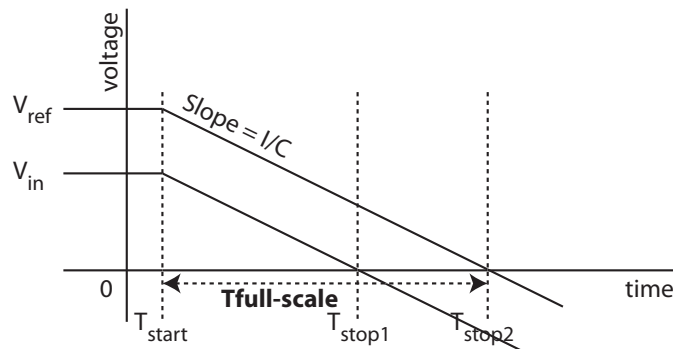


**Fig. 22:** Waveforms and important timing points in the VTC, if the oscillator is tuned to make $T_{stop2}$-$T_{start}$ equal to $T_{full\text{-}scale}$.

Another solution is to let the reference slope run for $T_{full\text{-}scale}$ and retain the end voltage, $V_{end}$. If the input range from 0 to $V_{ref}$ is 'compressed' into the range $V_{end}$ to $V_{ref}$, the input range is mapped onto the TDC range. This situation is depicted in figure 23.
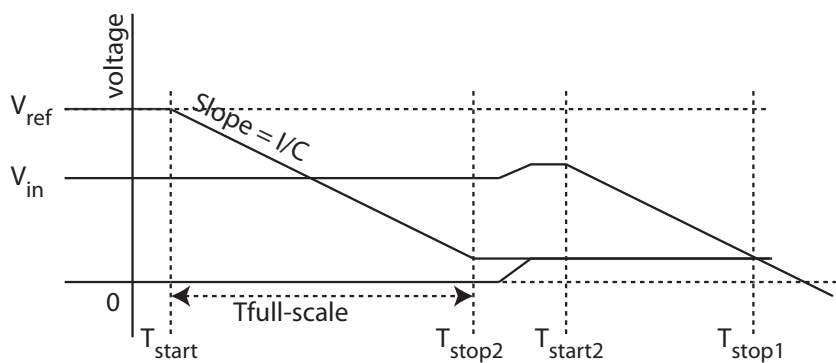


**Fig. 23:** Waveforms and important timing points in the VTC, if the input range is scaled to use $T_{full\text{-}scale}$.

In this work, the first option is used: a digital division of $T_{stop2}$ and $T_{stop1}$. Division will be performed off-chip, as well as generation of the reference voltage, to keep the prototype simple and versatile. This way, also the absolute values of $T_{stop1}$ and $T_{stop2}$ will be available, and experiments can be done using different types of rounding after difision.

This option does have some drawbacks: Ffirst, digital division is a sequential process, and therefore not trivial to implement. Second, both $T_{stop1}$ and $T_{stop2}$ need to be digitized using the TDC. Third, the accuracy after division and rounding has to be sufficient for the desired number of bits. This at least requires $T_{stop2}$-$T_{start}$ to be guaranteed to be larger than $T_{full\text{-}scale}$. A more thorough investigation of the implications of digital division may be necessary, but was not performed within this thesis.

## 3.4  System overview

Figure 24 shows an overview of the proposed system, with all architectural choices incorporated, except for the reference sampling mechanism, this is omitted for clarity.

The time base of the TDC is shown in the bottom left. It consists of the oscillator, its output buffers and the coarse counter. The section surrounding it, inside the dashed L-shape, is one VTC 'channel' and its associated part of the TDC.
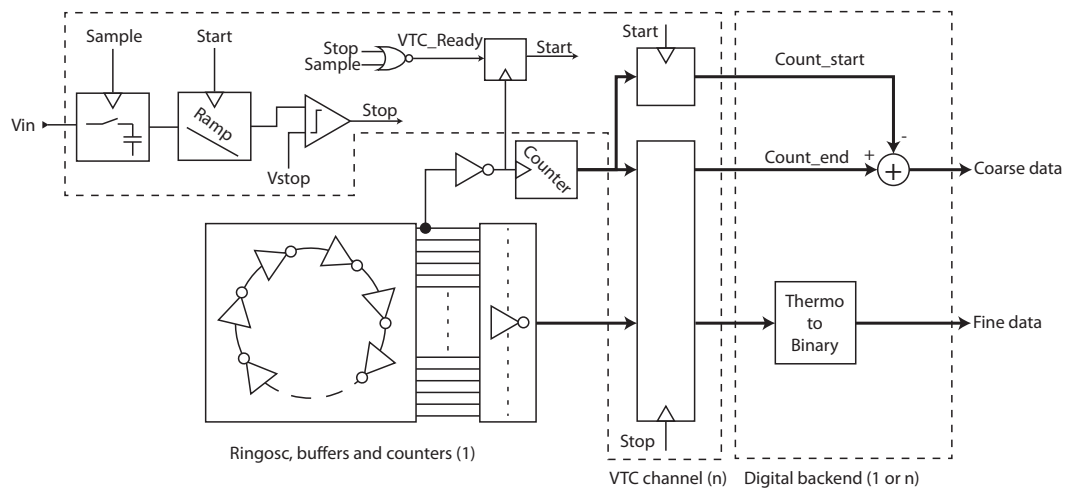
**Fig. 24:** Architectural overview of the proposed system.

It is assumed that the sample is taken on the falling edge of the sample clock. If both the sample clock and the 'stop' signal are low, indicating a sample has been taken, but the conversion still has to take place, the VTC generates a 'VTC_ready' signal. This signal is synchronized to the oscillator by a flip-flop to form a 'Start' signal. On this signal, the output of the coarse counter is registered and the VTC starts the ramp.

Once the ramp crosses the threshold, the 'Stop' signal is generated by the VTC. This signal is used to record the coarse counter output and the state of the oscillator.

The digital back-end subtracts the counter end value from the start value to yield the coarse bits. The state of the oscillator is decoded to binary. Since the stages in the oscillator were assumed a power of two, the fine bits can simply be appended to the coarse bits, and no adjustments are needed to the coarse or fine bit values.

### 3.4.1  Metastability

The architecture as shown in figure 24 suffers from two metastability issues.

The first is when the 'VTC_ready' signal violates the set-up and hold requirements of the flip-flop that generates the 'Start' signal. To solve this, the single flip-flop is replaced by a standard two-flip-flop synchronizer.

The second issue occurs when the stop edge arrives. Possibly, the measured state of the oscillator is a low value (e.g. 0 or 1), indicating that the coarse counter should have just been incremented. However, maybe the coarse counter has not noticed that 'edge' yet. Then, an error of one coarse bit is generated.

Two solutions are known for this issue: in [8], the coarse counter is decoupled from the ring and a regenerative latch regenerates the last known value at the counter clock input. This value is used to determine if the counter has just counted (and therefore the fine bits should be low), or if the counter is about to count (and therefore the fine bits should be high). However, decoupling the coarse counter from the ring means that one counter is required for every VTC channel.

Another solution is given by [14]. This work uses two counters on opposite phases of the oscillator. The fine bits are used to select the coarse counter that should be stable. A small post-correction to the chosen value is required, based on the fine bits. This solution is much more appropriate for our purposes, since the two counters can be re-used for every VTC channel.

Figure 25 shows the system overview again, now incorporating the two measures against metastability.
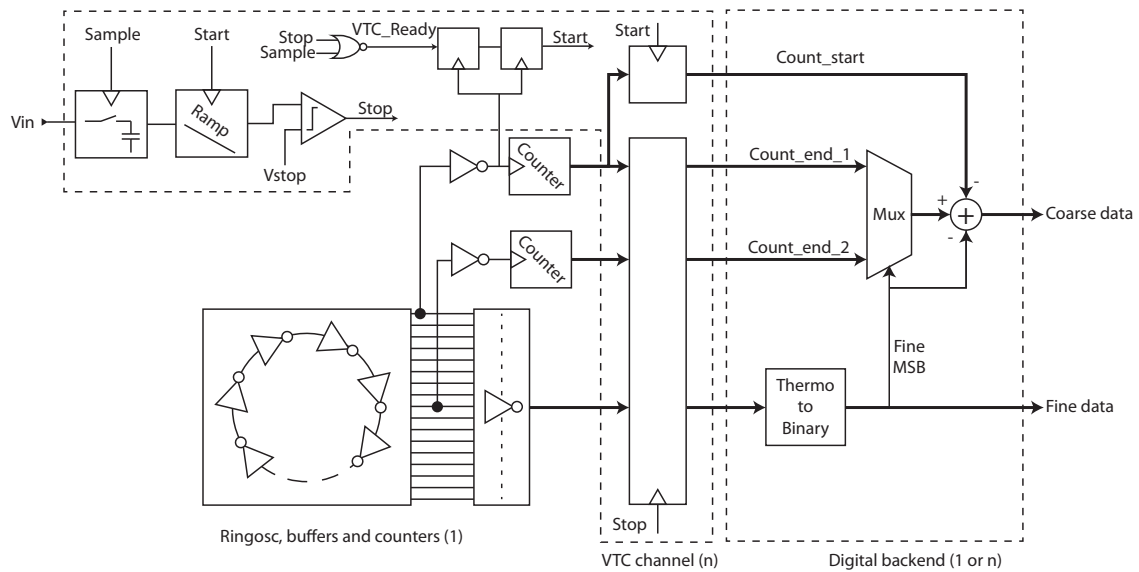
**Fig. 25:** Architectural overview of the proposed system with added measures against metastability.

## 3.5 Target specifications

To implement a proof of concept, a target resolution had to be chosen. Based on the TDC alone, many bandwidth-resolution combinations can be targeted, so the choice needed to be based on what is achievable with the chosen VTC topology.

Since the 'sample-and-ramp' VTC-topology was chosen because the linearity of V-to-I converters is generally limited to 6 bits or so, it makes no sense implementing a converter for 6 bits or less. On the other side, choosing more than 12 bits requires excessively large sampling capacitors [42], which negates the advantage of the ADC core being very digital and small.

Therefore, a resolution of 10 bits was targeted. To exploit the reconfigurability of the time-based ADC concept, one 10-bit VTC channel will be constructed such that it can also operate as multiple interleaved channels at lower resolution.

# 4   Implementation of the analog-to-time-to-digital converter

This section describes the steps taken toward a transistor-level proof-of-concept design of the ADC.

Despite the fact that the ADC concept was designed to maximally benefit from newer CMOS technologies, the proof of concept was implemented in NXP's mainstream 140 nm CMOS process.

The design of the TDC and VTC are highly intertwined, making it difficult to fully separate the discussion of their implementations. Nonetheless, first the implementation of the TDC will be discussed, followed by the implementation of the VTC.

## 4.1   TDC

While determining the system level architecture, the choice was made to implement the TDC as a flash ring TDC that operates uninterrupted and asynchronous to the sampling process. This reduces the design of the TDC to a ring oscillator with appropriate flip-flops or latches to determine its state, and some digital post-processing.

### 4.1.1   Ring oscillator

As mentioned during the system-level design, an oscillator is desired of which the number of stages is a power of two, implying an even number of stages. Such oscillators with an even number of stages generally have a stable latch-up state, which has to be suppressed to achieve oscillation. Several oscillator types exist that have an even number of stages, mainly because they are useful for quadrature signal generation. Most frequently used are current-mode logic oscillators and standard CMOS inverter oscillators with weak cross-coupled latches.

Current-mode logic (CML) oscillators have some advantages, such as high power-supply rejection [43]. However, they are not very suitable here because the stages draw a continuous current. In our topology, quite a lot of stages are needed to slow the oscillator down to a suitable rate for the coarse counter, so a CML oscillator would consume a lot of static power. Also, CML oscillators have low-swing output signals, which are not convenient in a highly digital system.

Simple CMOS inverter rings with weak cross-coupled inverters [43] do have full-swing outputs and consume little static current per stage. One problem with these oscillators is that the sizing of the weak inverters relative to the main inverters is rather critical: strong inverters make the oscillator power-hungry, and weak inverters do not prevent latch-up.

A more suitable oscillator was found in a front-end for ultra-wideband radio [44]. A schematic of one stage is shown in figure 26. Its inner workings are quite straightforward: MP1/MN1 and MP2/MN2 form two inverters. MP3/MN3 and MP4/MN4 prevent the outputs of the stage from changing if the inputs are not in antiphase, thus ensuring oscillation. This oscillator stage does not consume static power and has full-swing outputs. Under normal operating conditions, the cross-coupled transistors can be considered cascode devices, which is beneficial for noise and power consumption.

Early performance estimates of the oscillator and counter showed that an 8-stage, 16-phase oscillator would be the smallest power of 2 to leave sufficient timing margin for the counter to operate. Therefore, an oscillator of 8 times the depicted stage was adopted. The 16 output phases of the oscillator were buffered using unit inverters from the digital library.

As a starting point for the size of the transistors in the oscillator stage, all NMOS transistors were sized equal to those inside a minimal digital gate with two stacked NMOS transistors, such as a NAND gate, i.e. $0.856/0.16\,\mu\mathrm{m}$. This was considered a good starting point for a small, power-efficient oscillator, since the oscillator stages can also be considered digital gates with a limited fan-out: each stage will drive the next stage and a minimal inverter used as an oscillator output buffer. The PMOS transistors were also chosen minimum length, and their width was swept while observing the oscillator phase noise. From a W/L of $2.6/0.16$ upward, no further improvement in phase noise was observed, so this width was kept. Increasing either the four transistors M1-2 or the four transistors M3-4 while keeping the others at the same size did not improve the phase noise performance of the oscillator any further. The resulting phase noise after the output buffers is shown in figure 27.
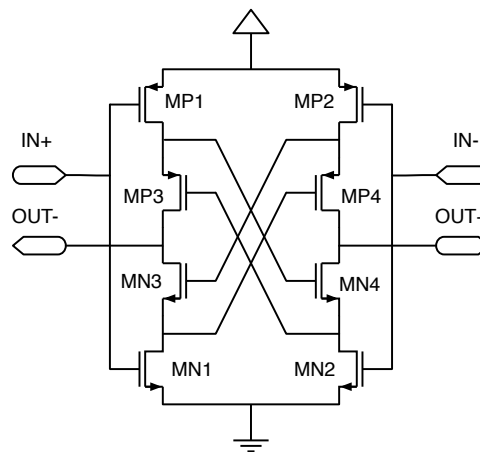
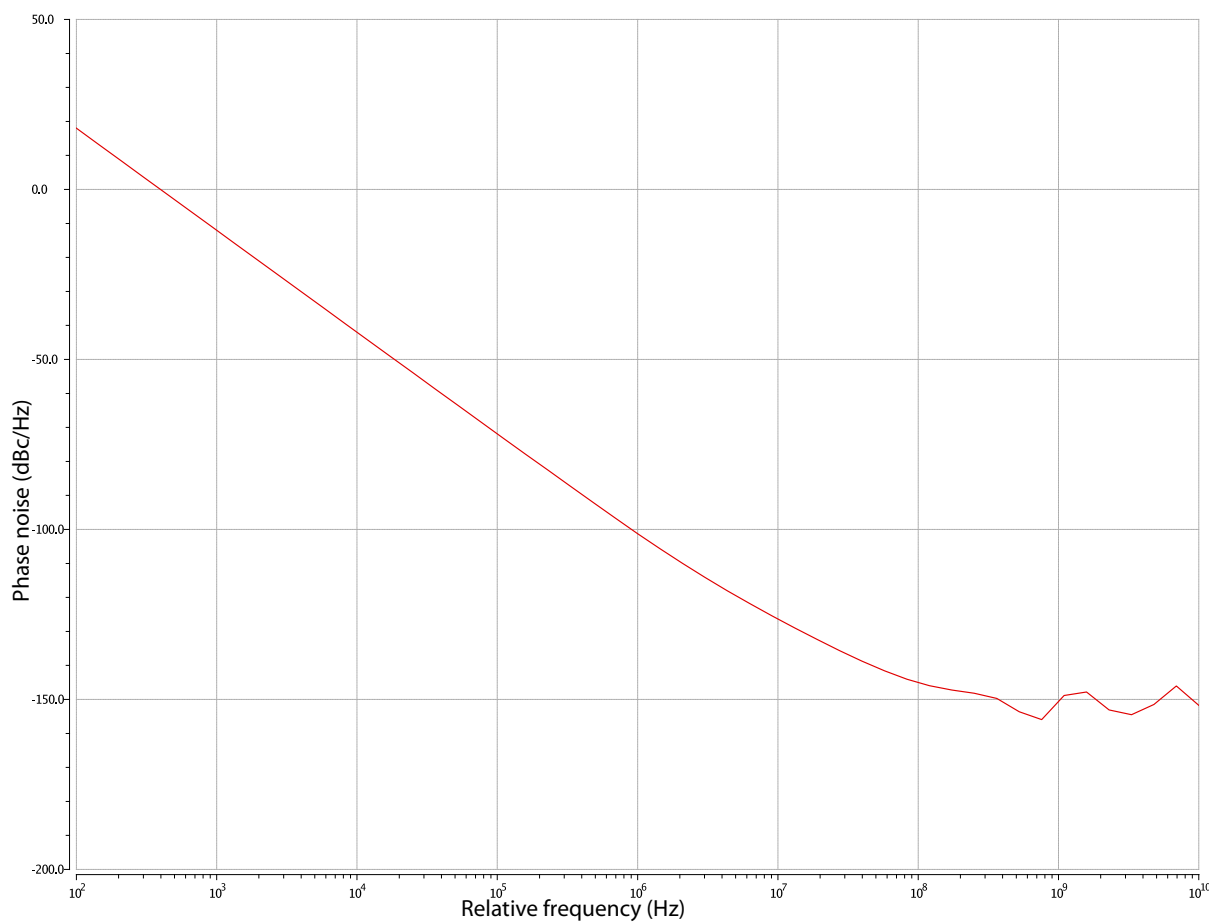**Fig. 26:** Schematic of the differential oscillator stage.



**Fig. 27:** Phase noise plot of the optimized 16-phase, 8-stage oscillator, with unit inverters as output buffers and no further loading, measured after the output buffers.

To verify if the oscillator is power-efficient compared to other ring oscillators, the oscillator FoM was calculated, defined as [45]

$$\text{FoM} = \mathscr{L}(f_{\text{f-2}})\left(\frac{f_{\text{f-2}}}{f_{\text{osc}}}\right)^2 \frac{P_{\text{core}}}{1\text{mW}} \tag{1}$$

where $f_{\text{osc}}$ is the oscillator frequency, $f_{\text{f-2}}$ is an offset frequency from $f_{\text{osc}}$ where the upconverted thermal noise dominates, $\mathscr{L}(f_{\text{f-2}})$ is the phase noise at this offset frequency, and $P_{\text{core}}$ is the power consumption of the oscillator core. Good ring oscillators come within 6 dB to the theoretical FoM limit, which lies at -165dBc/Hz at a temperature of 290K [46]. Simulations show that the oscillator designed above oscillates at 689 MHz with a phase noise of -126 dBc/Hz at 10 MHz offset, while the core (excluding output buffers) consumes 475 uA from the 1.8V supply. This results in a FoM of -164 dBc/Hz, which is very close to the theoretical limit, indicating that the oscillator design is good, even if some dB's of performance are lost due to parasitics later. The oscillator core achieves a nominal propagation delay of $t_{\text{d,nom}} = 90.7$ ps per stage, and requires 78 fJ per propagation step. This is an indication of the achievable time resolution and FoM of the TDC, however, both specifications will worsen when parasitics are included. Furthermore, the power consumption in the proposed architecture will be divided over the number of VTC channels that will be using the TDC simultaneously, resulting in a division of the FoM by the number of channels.

The variance in delay of a single stage due to thermal noise can be calculated from the phase noise in the thermal region [47]:

$$\sigma_{\text{td,thermal}} = \frac{f_{\text{f-2}}}{f_{\text{osc}}} \cdot 10^{\frac{\mathscr{L}(f_{\text{f-2}})}{20}} \cdot \sqrt{t_{\text{d,nom}}} \tag{2}$$

resulting in $\sigma_{\text{td,thermal}} = 69$ fs, or about 0.077%. This is equal to the thermal period jitter of one oscillator output phase, divided by $\sqrt{16}$, since the oscillator has 16 stages.

In section 2, the statement was made that generally, the effects of thermal jitter generally lie about an order of magnitude below those of mismatch jitter. To verify that statement for the oscillator at hand, mismatch simulations were also executed. A 100-point Monte-Carlo simulation of the oscillator showed a standard deviation in the delays of $\sigma_{\text{td,mismatch}} = 2.43$ps or 2.7% (excluding mismatch in the output buffers). Therefore, for this specific oscillator stage, the effects of thermal noise lie about a factor 35 below those of mismatch.

As demonstrated in section 2, using a ring TDC turns mismatch errors into a cyclic INL pattern of a low magnitude. The remaining INL profile is comparable to that of a short delay line of 16 stages, repeating itself. The standard deviation of the maximum point of this INL pattern can be estimated by [48]:

$$\sigma_{\text{INL,max}} = \sqrt{\frac{N}{4}} \frac{\sigma_{\text{td,mismatch}}}{t_{\text{d,nom}}} \tag{3}$$

Inserting N= 16 stages, $\sigma_{\text{td,mismatch}} = 2.43$ps and $t_{\text{d,nom}} = 90.7$ps, $\sigma_{\text{INL,max}} = 0.054$LSB. Therefore, the mismatch jitter will not limit the performance of this TDC, and the number of bits that can be calculated becomes dominated by the thermal jitter. A quick calculation shows that an edge should propagate through about 430000 of these stages to develop half a unit delay (half an LSB) of thermal jitter. In other words, theoretically a TDC of more than 18 bits could be constructed using the proposed oscillator, however, at such lengthy propagation times, other (low-frequency) effects will start influencing the propagating edge, such as supply voltage and temperature drift and flicker noise.

**Timing**

Since the TDC will be running asynchronously, measures need to be taken to guarantee sufficient accuracy of the converter under all process, voltage and temperature conditions. At slow conditions for the oscillator, a TDC LSB becomes longer, so the TDC can calculate less levels per second. Therefore the amount of time the TDC needs to make $2^{10} = 1024$ time steps under the slowest conditions dictates the full-scale output of the VTC.

For reasons that will be detailed in section 4.2.1, the VTC will be configurable to act as either one channel with 10-bit performance or four channels with 9-bit performance. In this 9-bit mode, the full-scale time from the VTC should cover 512 of the slowest possible time steps from the TDC.

The slowest oscillation conditions are defined here as the slow process corner, with 10% reduced supply voltage and a temperature of 80 °C. Simulations show that under these conditions, the delay per stage increases to 147 ps. For the calculation of 10 bits, the full-scale input time would then be $1024 \cdot 147$ ps $= 150.5$ ns. For the 9-bit mode, the full-scale time is 75.3 ns.

Besides time for the ramp, the time required for conversion should also include the 16 to 32 time steps required by the synchronizer to generate the 'Start' signal from the the 'VTC_ready' signal (refer to figure 25). At the same slow conditions, this equals 2.35 - 4.70 ns. Finally, some margin has to be included for additional delay from the comparator toggling to the instant the oscillator state is sampled.

The prototype will be clocked by the same clock signal in both 10-bit and 9-bit modes. Table 3 illustrates how the converter operates from this clock signal in both modes. In 10-bit mode, the VTC requires one clock cycle for sampling and two for slope conversion, whereas in 9-bit mode, all four channels require one clock cycle for sampling and one for slope conversion.

For 10-bit conversion, the slope phase takes 150.5 + 4.7 = 155.2 ns plus margin, for 9-bit conversion it takes 75.3 + 4.7 = 80 ns plus margin. Therefore, a master clock of 11 MHz will be used, resulting in cycles of 90.9 ns, giving 10.9 ns margin in 9-bit mode and 26.6 ns margin in the 10-bit mode.

**Tab. 3:** Timing of the converter when the VTC is operating as a 10-bit channel or as four interleaved 9-bit channels.

| Cycle | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 10-bit | Sample Vin | Slope | | Sample Vref | Slope | | |
| 9-bit, ch1 | Sample in | Slope | Sample ref | Slope | | | |
| 9-bit, ch2 | | Sample in | Slope | Sample ref | Slope | | |
| 9-bit, ch3 | | | Sample in | Slope | Sample ref | Slope | |
| 9-bit, ch4 | | | | Sample in | Slope | Sample ref | Slope |

Besides the slowest oscillation conditions, the fastest oscillation conditions are also important: these determine, together with the synchronizer, the minimum time between the 'VTC_ready' signal and the 'Start' signal for the slope, in other words how much time there is available for the VTC to prepare for the slope after the input sample is taken.

The fast oscillation conditions are defined as the fast process corner, combined with 10% increased supply voltage and a temperature of $-20$ °C. Under these conditions, the propagation delay of a stage is only 64.8 ps, corresponding to an oscillation frequency of 965 MHz. In case of a two-flip-flop synchronizer, the VTC has at least one oscillation period available, or about 1 ns. Should this be too little, then the synchronizer can be extended with more flip-flops, each one adding one oscillation period of about 1 ns to the time the VTC has available. Care has to be taken that this time does not exceed the margins that were included in the sampling period under the slowest oscillation conditions.

### 4.1.2  Digital backend

To register the state of the oscillator and counters, and translate the results into a binary code, appropriate flip-flops and some digital circuitry are required. The implementation of these blocks is discussed next.

**Flip-flops**

Since the outputs of the oscillator are 16 full-swing signals, they could theoretically be sampled using flip-flops from the digital library. To find out if this is indeed possible or if more sophisticated, custom flip-flops need to be designed, the expected shortcomings of library flip-flops were listed for further investigation:

- Metastability
- Mismatch of the input threshold
- Mismatch in timing path from clock to sampling instant

- Kickback
- Loading of the oscillator
- History effects
- Power supply rejection

Metastability is not expected to be an issue: although the flip-flops sample a high-frequency signal on a critical clock edge, they are given a significant portion of the (much slower) sampling period to regenerate their outputs afterwards.

To find out if mismatch would cause trouble, the digital library flip-flops were put on a test bench, depicted in figure 28, and subjected to Monte-Carlo simulations while sampling a full-swing transition. The test bench uses library inverters to provide realistic transitions on the clock and data inputs of the flip-flop. The clock input is an unfiltered transition from a unit inverter, whereas the data input is an edge that is slowed down using some capacitance, to have a rise time of about 90 ps. This signal imitates the unbuffered output of the oscillator, and serves as a worst-case signal for the flip-flop to sample. Mismatch contribution of the inverters and capacitors was excluded from the simulation, to ensure that only the effect of the flip-flop itself is studied.
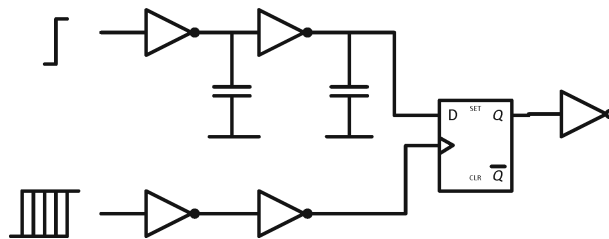


**Fig. 28:** Test bench used to test the mismatch of library flip-flops. Only the flip-flop itself is set to contribute mismatch.

The result is shown in figure 29. A cumulative distribution function was fit to the results using MATLAB, to determine the standard deviation, which turned out to be 3.19 ps, or about 3.5% of the nominal delay. In other words, using library flip-flops in the TDC would contribute about 0.035 LSB RMS to its DNL, so no problems are expected due to poor matching of library flip-flops.

To check if kickback and loading would cause trouble, a bank of flip-flops was connected to the oscillator. Three output periods of the oscillator were compared: one where the flip-flops were 'tracking', one where the flip-flops were switched from 'tracking' to 'holding' halfway the period, and one where the flip-flops were 'holding'. There was no noticeable deviation in length between these three periods, indicating sufficient isolation between oscillator and flip-flops. If isolation problems are expected after including parasitics, the isolation between oscillator and flip-flops can always be improved by using two or more cascaded buffers.

History effects were not studied extensively, but can always be prevented by using flip-flops with a reset input, if necessary. Power supply rejection of library flip-flops was also not studied further, but may be worth investigating.

All in all, no severe problems are expected when simple library flip-flops are used in the TDC.

**Decoding logic**

The output of the oscillator has the form of a 'travelling pulse': at any given time, half of the outputs will be logic high, and half will be logic low. This pattern cycles through the oscillator. To decode such a pattern into the fine binary code, the structure of figure 30 is proposed. It consists of mostly inverting logic, since such gates are smaller and faster than non-inverting gates.

A bank of NAND gates with one inverting input is used to find the transition from 0 to 1. This results in a zero at the location of the transition, comparable to 'one-hot' coding. In normal one-hot coding, a tree of OR-gates is used to convert the position of the '1' to a binary representation. In this case, a variation of the OR-tree is used, consisting of only inverting gates, to convert the position of the '0' to binary. Because of the inherent monotonicity of the TDC, no bubble correction is expected to be necessary.

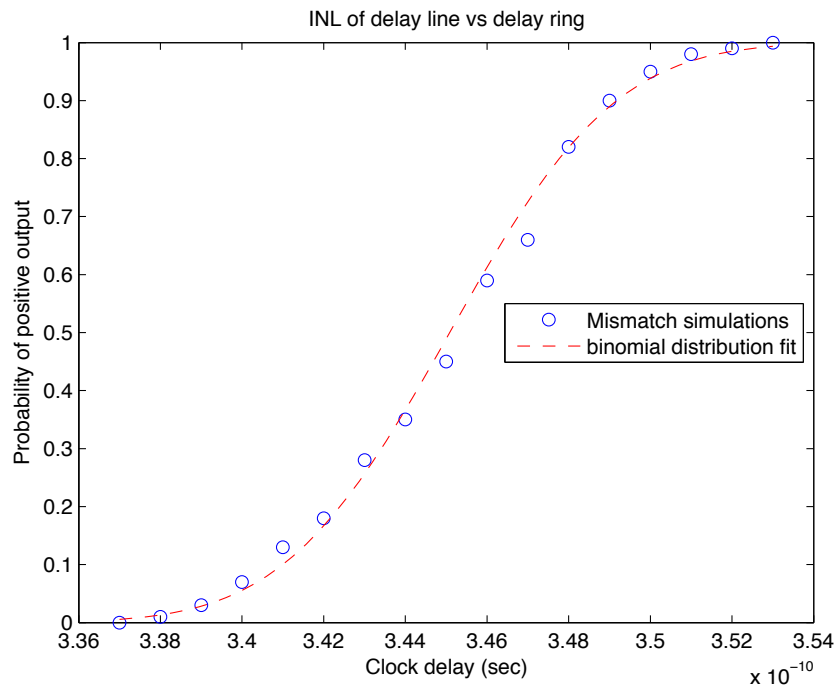**Fig. 29:** Probability that a flip-flop will sample an edge as a positive input, as a function of a delay in the clock signal, under influence of mismatch. The absolute value of the clock delay is arbitrary. For each point, 100 Monte-Carlo runs were executed.
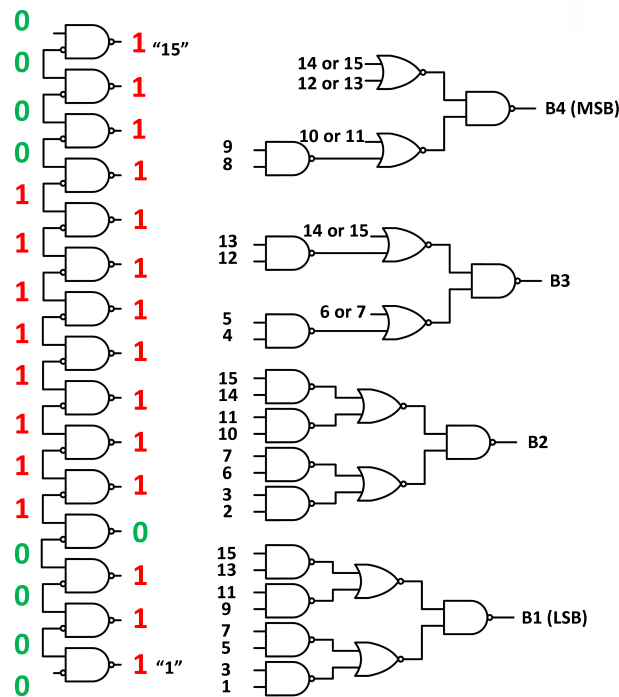


**Fig. 30:** Decoding logic to extract fine binary code from the oscillator state. The signals 'X or Y' are outputs of another NAND gate combining X and Y.

**Counter**

To keep track of the coarse bits, two synchronous counters are required on opposite output phases of the oscillator. The fine bits are used to select the counter that has a stable output, thus preventing metastability.

If the TDC full scale would always be 1024 time steps, or 1024/16 = 64 oscillation periods (in 10-bit mode), a 6-bit counter would be sufficient. However, this is only true for the slowest oscillation conditions. To cover also the faster oscillation, the counter needs some additional bits. Since the fastest conditions are slightly more than twice as fast as the slowest conditions, two extra bits are needed.

The 8-bit counters were implemented as shown in figure 31. The counters have a reset input, which is needed to synchronize them when the converter is started. Synchronized counters are required because if counter 1 toggles from e.g. 50 to 51 at a certain edge from the oscillator, counter 2 also has to toggle from 50 to 51 half an oscillation period later to obtain a coherent range of output values.

The proposed counter circuit was tested to function correctly at close to 2 GHz under nominal process conditions. It consumes on average 232 μW when clocked at the nominal oscillation frequency of 689 MHz.
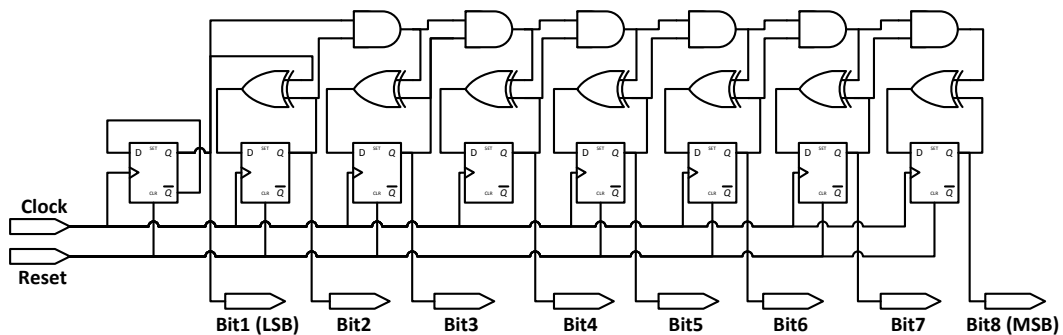


**Fig. 31:** Implementation of the 8-bit synchronous counter.

**Remainder of the digital backend**

The remainder of the digital backend consists of:
- selecting the stable counter based on the fine code
- correcting the value based on the MSB of the fine code
- subtracting the initial counter value to obtain the digital output
- performing digital division of the input sample by the reference sample

These blocks were not implemented further: the proposal is to let the proof-of-concept output the 'raw data' from the flip-flops and thermo-to-binary decoder and perform the listed post-processing steps off-chip.

## 4.2  VTC

This section discusses the step-by-step implementation of the voltage-to-time converter. First, a start-voltage controlled VTC mechanism is proposed and after the dominant noise sources have been identified, the components are sized.

### 4.2.1  Sample and ramp circuit

To generate a very linear ramp on a capacitor, a high-impedance cascoded current source can be constructed [38–40], but this is not a very technology-scaleable solution. Feeding the current into a virtual ground node puts much

less requirements on the current source. However, constructing a complete op-amp to implement a virtual ground is not a very portable solution either.

The proposed solution is to sample the input voltage on a capacitor, then connect the capacitor across a single-transistor OTA. This OTA provides a virtual ground for the current source, but also re-uses the current through the capacitor to bias itself. This technique has recently been succesfully applied in a low-power relaxation oscillator [46]. The mechanism is illustrated in figure 32.
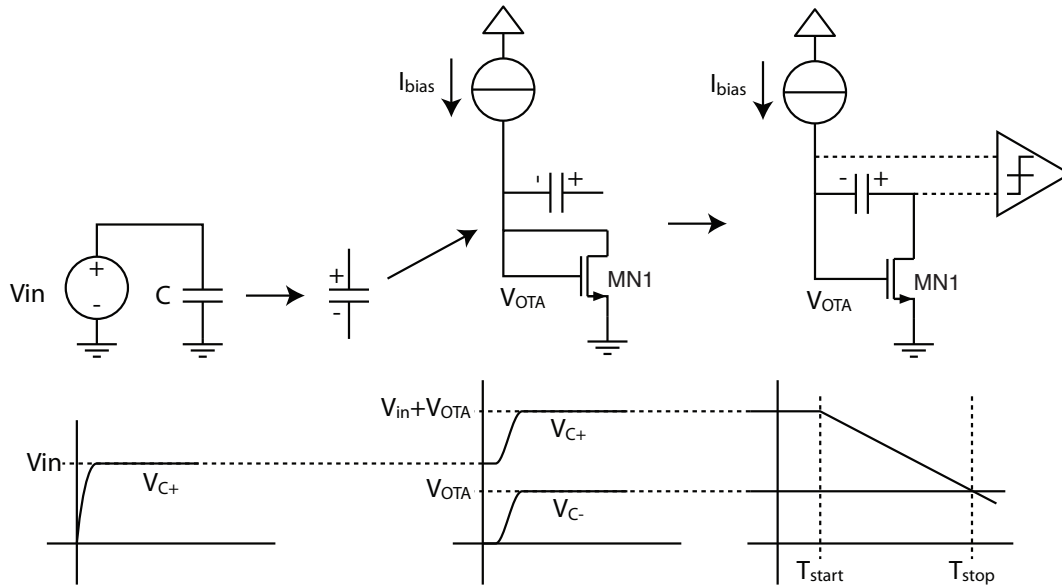


**Fig. 32:** Operation principle of the VTC.

During the sampling period, the OTA transistor MN1 is connected as a MOS diode. The diode voltage is also the virtual ground voltage of the OTA, or $V_{OTA}$. After sampling, the capacitor is stacked onto this diode voltage and given some time for the voltage levels to shift. Now, the VTC is waiting to start the linear ramp, and on the 'Start' signal from the TDC, the diode is opened, the capacitor is connected across the OTA and the bias current is redirected through the capacitor. The zero-crossing of the voltage across the capacitor is detected by a comparator to generate the output signal.

**Input range**

The proposed sample and ramp circuit allows for a wide input range of the converter, theoretically the supply voltage minus $V_{OTA}$. Since 140nm CMOS uses a 1.8V supply and has threshold voltages of about 400 - 500 mV, a wide input range of 0V to 1V single-ended was chosen, leaving plenty of room to select a suitable $V_{OTA}$. If desired, the input range could be extended to a rail-to-rail input by capacitive division of the sampled value, before the slope conversion is started.

**Sampling capacitor**

The capacitor should be a fringe capacitor for the best trade-off between density and linearity. To size the capacitor itself, the noise performance of the proposed circuit has to be analyzed. Flicker noise sources are treated elsewhere, since the reference sampling mechanism compensates for most of their low-frequency effects on system level. The relevant sources of thermal noise in this circuit are the sampling noise, the noise from the current source and the noise of the comparator. The OTA will not be a dominant source of noise, since its noise will be suppressed by its own loop gain.

The voltage uncertainty due to the sampler noise is given by the well-known equation:

$$\sigma_V^2 = \frac{kT}{C} \tag{4}$$

where $k$ is the Boltzmann constant, $T$ the absolute temperature and $C$ the size of the sampling capacitor.

The current source will be implemented as a MOSFET with a degeneration resistor. Sub-micron MOSFETs in strong inversion are always noisier than resistors by at least a factor $2\gamma$ [46], with $\gamma$ being the noise excess factor, typically between 1 and 1.5. Therefore, a best-case value for the noise from the current source can be calculated by considering it a resistor [46]. The uncertainty after integrating the noise from the current source into the capacitor for a time $T_{\text{on}}$ is then given by

$$\sigma_V^2 \geq \frac{2kT}{C^2} \frac{I}{V_{\text{cur}}} T_{\text{on}} \tag{5}$$

where $I$ is the magnitude of the current and $V_{\text{cur}}$ the voltage headroom reserved for the current. Now, since the voltage over the capacitor has to be reduced from the single-ended input voltage $V_{\text{in}}$ to zero within $T_{\text{on}}$ we can write $V_{\text{in}} = T_{\text{on}}(I/C)$ and therefore

$$\sigma_V^2 \geq \frac{2kT}{C} \frac{V_{\text{in}}}{V_{\text{cur}}} \tag{6}$$

So the noise is not dependent on the actual integration time, just on the voltage drop that needs to be created on the capacitor and some design choices. Adding the two noise sources so far:

$$\sigma_V^2 \geq \frac{kT}{C}(1 + \frac{2V_{\text{in}}}{V_{\text{cur}}}) \tag{7}$$

Comparator noise is discussed later; for now, some margin is kept in the design for additional noise from the comparator.

Note that the noise contributions from sampling and the comparator are signal-independent, but the noise from the ramp itself is signal-dependent: For low $V_{\text{in}}$, the noisy current source is only integrated for a short time and therefore hardly contributes noise; for $V_{\text{in}} = V_{\text{full-scale}}$ it is integrated during the full-scale time and generates the worst case noise, given by

$$\sigma_V^2 \geq \frac{kT}{C}(1 + \frac{2V_{\text{full-scale}}}{V_{\text{cur}}}) \tag{8}$$

Equating the noise of the VTC to that of an ideal B-bits quantizer, we obtain [42]:

$$\frac{kT}{C}(1 + \frac{2V_{\text{full-scale}}}{V_{\text{cur}}}) = \frac{\Delta^2}{12} \tag{9}$$

where $\Delta$ is the size of an LSB in volts, or $\Delta = V_{\text{full-scale}}/2^B$ with $B$ the number of bits in the converter. This can be rewritten to

$$C = 12kT(1 + \frac{2V_{\text{full-scale}}}{V_{\text{cur}}})(\frac{2^B}{V_{\text{full-scale}}})^2 \tag{10}$$

so to obtain the smallest possible capacitance, a high voltage headroom is desired, as well as a high $V_{\text{full-scale}}$.

A proof-of-concept is preferably not limited by thermal noise, in order to better study the performance of the topology itself: its tonal performance, linearity et cetera are more easily measured when the thermal noise performance is better than the quantization noise performance. To have some margin for noise in the comparator and TDC, a thermal noise performance of 12 bits was targeted for the sample and ramp circuit. Given that the voltage headroom for the current source will be about 1.8V - $V_{\text{OTA}}$ = 1.2V, and the input voltage range will be about 1V single-ended, a capacitor of 2.22pF is required.

**Operation in two modes**

As mentioned before, the VTC will be operating in two modes, to demonstrate the reconfigurability of the analog-to-time-to-digital concept: either one channel with 10-bit performance, or several interleaved channels with 9-bit performance can be selected. From equation 10 it becomes apparent that for one bit less of noise performance, the sampling capacitance may be divided by four. Therefore, the sampling capacitance of the designed 10-bit VTC can simply be subdivided in four equal parts to form four 9-bit channels. These parts are connected in parallel using MOS switches when the channels are operating in unison (10-bit mode). The resulting schematic is shown in figure 33.

Four channels of the proposed VTC can be distinguished. Switches 'Sp$i$' and 'Sn$i$' connect the positive and negative poles of the sampling capacitances together for operation in 10-bit mode. The function of the other switches, in the order in which they are relevant within a sampling period, is as follows:

- S$i$a are the sampling switches for the input voltage.

- S$i$b are the sampling switches for the reference voltage.

- S$i$c disconnect the bottom plate from ground so the sampling capacitor can be stacked on top of the OTA. Since this switch is required anyway, bottom-plate sampling can be readily applied to prevent signal-dependent charge injection from the sampling switches.

- S$i$d connect the bottom plate of the capacitor to the OTA input.

- S$i$f are the switches that normally keep the OTA in diode mode while the input is tracking or level-shifting. After level shifting, the VTC awaits the start signal from the TDC. At the start signal, S$i$f is opened and S$i$e is closed, incorporating the capacitor in the loop of the OTA and starting the slope conversion.

The function of switches S3g and S4g is to turn off the bias current sources I3 and I4 in 10-bit mode. This can be understood by considering the following: For 9-bit accuracy from the TDC, only half the conversion time is required compared to the 10-bit mode. This fact allows the interleaving method that concluded in the 'timing' paragraph of section 4.1.1. Therefore, in 9-bit mode, a quarter of the 10-bit sampling capacitance will have to be discharged in half the time. For this to happen, each of the four channels requires half the discharge current from the 10-bit mode. This implies that two more current sources are needed.

To keep the characteristics of the OTA transistors equal in both modes, S3d and S4d will always remain open in 10-bit mode. So when only half of the bias current is fed into the VTC, only half of the OTA transistors will be used.

**Assisted level-shifting**

One problem of fringe capacitors is that if the lowest metal layer is included, they have a large parasitic capacitance from both pins to the shielding well on which they are placed. To perform the level-shifting to stack the capacitor onto the OTA, the parasitics on both terminals need to be charged with an additional $V_{OTA}$, if the well is grounded. This takes a long time when only the bias current is used to charge them.

The proposed solution is to assist the level-shifting by switching the well below the capacitor from ground to $V_{dd}$, just after the capacitor is connected to the diode-connected MN1, as illustrated in figure 34. MN1 only has to prevent the level shift from overshooting by draining excess charge from the parasitics, which is much faster than the previous solution of completely charging the parasitics using the bias current. For this to work, the capacitor needs to be placed on an N-well, which is allowed to be connected to any voltage.

Upon switching the well to $V_{dd}$, the parasitics will form a capacitive division with the load capacitances present on both ends of the capacitor. Therefore, it is important that the parasitics are large enough compared to e.g. the input capacitance of the comparator. If the parasitics are not large enough to lift the sampling capacitor to above $V_{OTA}$, the remainder of the level shift will have to be performed by the slower mechanism of the bias current charging the parasitics.

The PMOS switch pulling the well to $V_{dd}$ was sized $2/0.16\,\mu m$ and the NMOS switch was chosen minimum-size ($0.768/0.16\,\mu m$), based on transient simulations further along the design procedure.

There may be parasitics from the well to the substrate that are not included in the capacitor model. Also, if the parasitics to both plates are unequal or nonlinear, a signal-dependent differential mode charge may be injected onto
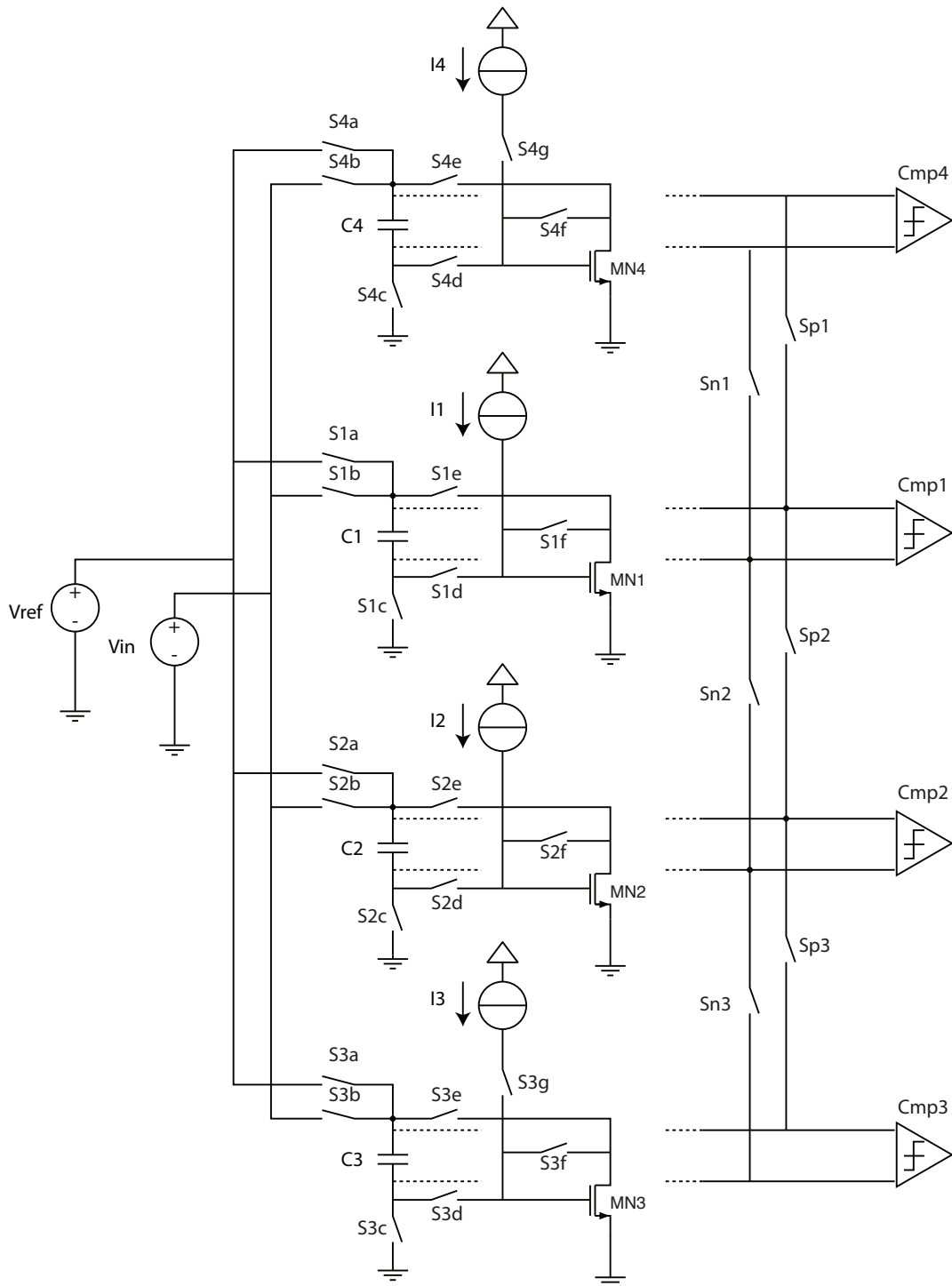
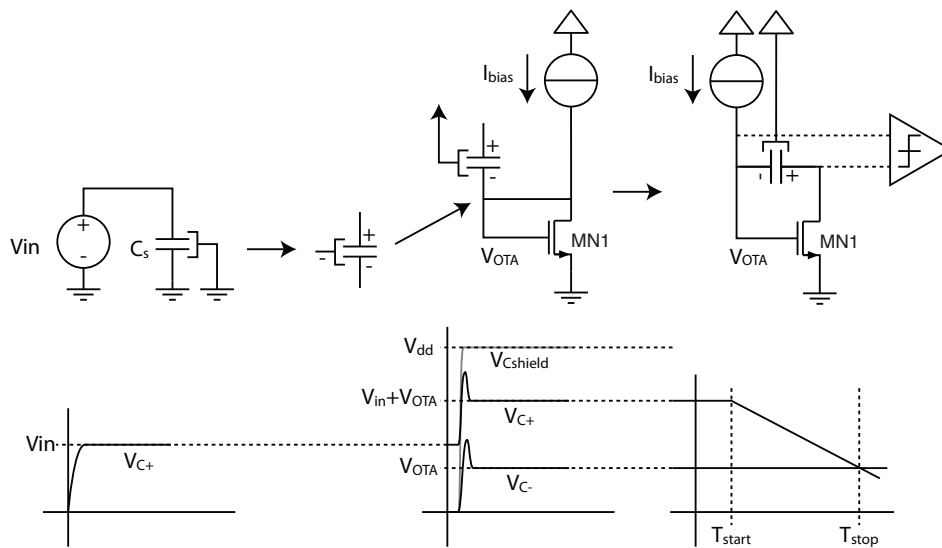**Fig. 33:** Schematic of the reconfigurable VTC.

**Fig. 34:** Mechanism of assisted level-shifting in the VTC.

the sampling capacitor. For these reasons, this level shifting mechanism may require attention from someone more experienced in this matter.

**Sampling switch and bottom-plate switch**

The on-resistance of the sampling switch is dictated by two requirements: For one, it should be low enough to allow a static input voltage to settle on the sampling capacitor to within the desired accuracy (usually half an LSB). For 10-bit accurate static settling, about 7.6 RC-time constants are needed [42].

However, apart from this static settling error, the RC-constant also acts as a low-pass filter when the sampler is tracking dynamic signals, causing some amplitude attenuation [42]. If the sampler is dimensioned at the afore-mentioned 7.6 RC-constants, a Nyquist input signal is attenuated by about 2%. At 10 time constants, the atten-uation reduces to about 1%. Therefore, to achieve some capabilities of sampling signals beyond Nyquist (i.e. sub-sampling), 10 time constants were targeted.

The sampling time is equal to one 11 MHz clock cycle in both modes, or 90.9 ns. So one RC-time should equal 9.09 ns, and since C = 2.22pF in 10-bit mode, R = 652 $\Omega$.

The proposed design requires a bottom-plate switch (S*ic* in figure 33). While sampling, the bottom switch and sampling switch are effectively in series, so their total resistance should not exceed the calculated 652 ohm. Each switch in 10-bit mode is essentially made up of the four switches from the 9-bit mode operating in parallel.

For the sampling switches, NMOS switches of W/L = 2/0.16 $\mu$m were selected, for a maximum resistance (at maximum input signal) of 1.35 k$\Omega$. The bottom-plate switches can be smaller because they can always be driven with the full supply. Therefore, NMOS switches of W/L = 1.5/0.16 $\mu$m were selected, yielding an on-resistance of 656 $\Omega$. The total on resistance in 10-bit mode (four sampling switches in parallel, in series with four bottom-plate switches in parallel) equals 502 $\Omega$.

Since the bottom-plate switch is required anyway, bottom-plate sampling is employed: The bottom switch is opened slightly before the sampling switch to prevent signal-dependent charge-injection from the sampling switch. However, this leaves the sampling capacitor floating while the sampling switch is opened. Opening of the sampling switch now injects charge from its channel as common-mode charge onto the sampling capacitor, pushing the sampling capacitor outside the supply rails. To prevent this, dummy NMOS switches of W/L = 1/0.16 $\mu$m are included on both sides of each sampling switch, and clocked with the complement of the sampling signal to absorb the charge.

**Ramp current source**

The ramp current has to discharge the sampling capacitor from a full-scale 1V sample to zero over a time span in which the TDC is guaranteed to calculate at least 10 bits. In section 4.1.1, this time span was determined from the slowest operation conditions to be 151ns. To generate a ramp of 1V magnitude over 2.22pF within this time span, a current of 14.7 μA is required. This current will be provided by two matched PMOS current sources of 7.35 μA, to allow easy activation of two more matched current sources in the 9-bit mode.

Periodically sampling a reference voltage to track the full-scale of the converter is expected to make the system insensitive to flicker noise in the current source, but only roughly up to the frequency with which the full-scale is converted. Therefore, it is a good idea to resistively degenerate the PMOS transistors, since resistors have far less flicker noise. The transistors then act like cascode transistors and the degeneration resistor will be the actual current source. However, degeneration absorbs some headroom, so the cascode transistor should not be generated so much that it enters the triode region. Also, degeneration decreases the source voltage of the transistor, causing the bulk effect to increase the threshold voltage.

For matching considerations, the current sources are chosen quite long so they can be biased at quite some overdrive (about 130 mV). So, the current source needs over 130mV $V_{ds}$ to stay saturated, while keeping about 600 mV of room for $V_{OTA}$ on the source side and $V_{dd} - I \cdot R$ on the drain side. So we can safely drop about 700mV across the resistor, at 7.35 μA this equals about 94 kΩ. The current source PMOSTs are chosen W/L = 5/0.5 μm, the N-poly degeneration resistor is chosen W/L = 0.16/80 μm for a resistance of about 82 kΩ. Monte-carlo simulations show that the resulting matching between channels is in the order of 1%.

Figure 35 shows three implementations of the 14.7 μA current source as it will be applied in 10-bit mode, for noise comparisons: a poly-resistor implementation, a PMOS implementation and a degenerated PMOS implementation. $V_{OTA} = 460$mV as a result of early simulations on the remainder of the system. All transistors were sized 5/0.5 μm, R2 = 0.16/80 μm and R1 = 0.16/88.5 μm. For both active implementations, the multiplier was set to $M = 2$ to emulate the two current sources in the VTC, for the resistor implementation this was included in the resistor value itself. The bias voltages were determined by DC simulations on a reference branch, and implemented as noiseless voltage sources.
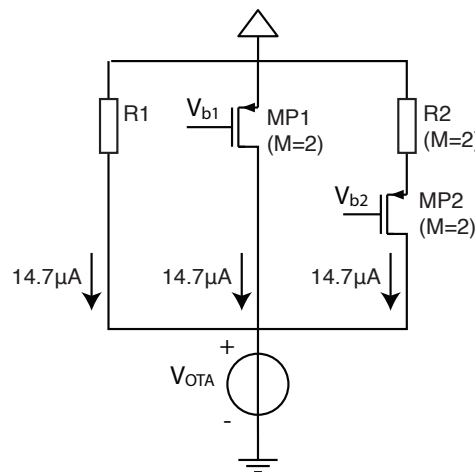


**Fig. 35:** Three implementations of the current source for noise comparisons. From left to right: a poly resistor implementation, a PMOS current source and a degenerated PMOS current source.

Figure 36 shows the results of noise simulations on all three implementations. In the thermal noise region, the MOS source shows a noise of 1.613 pA/$\sqrt{\text{Hz}}$, the degenerated MOS shows 0.592 pA/$\sqrt{\text{Hz}}$ and the resistor shows 0.427 pA/$\sqrt{\text{Hz}}$. For very high frequencies, the noise increases again due to parasitic capacitances lowering the effective noise impedance.

Degeneration clearly helps the active current source to approach the noise characteristic of a resistor. Since $\sigma^2/\Delta f = 4kTR_{noise}$, we can say that the degenerated MOS source is $(592 \cdot 10^{-15} A/\sqrt{Hz})^2/(427 \cdot 10^{-15} A/\sqrt{Hz})^2 = 1.92$, so still almost twice as noisy as a resistor. Put differently, its noise is equal to a resistor implementation with
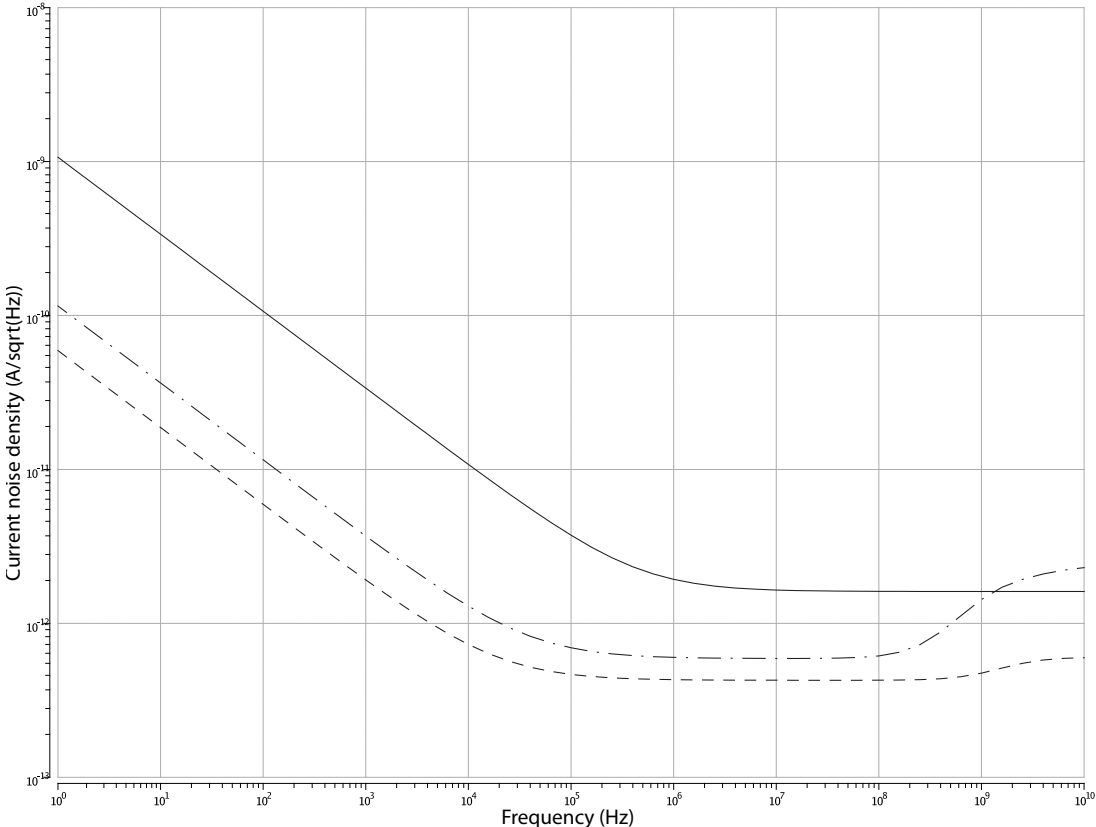
**Fig. 36:** Current noise spectra of three different implementations of the current source: a resistor (dashed line), a MOS source (regular line) and a degenerated MOS source with approximately half the voltage headroom consumed by the degeneration resistor (dash-dot).

half the voltage headroom. This will degrade the noise performance of the VTC in the slope phase, but since it was designed for 12 bits noise performance, there is some performance to spare. More details will follow in section 5.2.2.

The output resistance of the reference branch is $58\,k\Omega$. For low noise on the bias line from the reference current to the mirrored currents, the bias line is decoupled to the supply rail using a MOS capacitor of $20\,\mu m$ x $20\,\mu m$. The resulting capacitance of $3.7\,pF$ reduces the bandwidth of the noise from the reference branch to $742\,kHz$.

**OTA transistor**

An important factor in sizing the NMOS OTA transistors is that they provide a solid virtual ground to the current sources, independent of the voltage ramp that is present on the drain. This allows the current source to feed a constant current into the sampling capacitor, ensuring a linear ramp, resulting in good linearity of the entire converter. To provide a good virtual ground, the OTA needs to show a high output impedance.

Another consideration is that the overdrive of the OTA must not be too large, since a high $V_{OTA}$ will consume headroom that is required for the signal.

Guided by preliminary simulations, the OTA transistors were sized $5/0.5\,\mu m$. This resulted in a nominal $V_{OTA}$ of 460mV (420 mV threshold + 40 mV overdrive), with signal-dependent variations in the order of 10 mV. According to simulations, the current source has an output impedance of about $14.5\,M\Omega$, therefore 10 mV of variation on the virtual ground would result in a current variation of below 1nA. Since this is an order of magnitude below 1/1024 of the full current of $14.7\,\mu A$, no linearity problems were expected.

**Other switches**

Switches S$id$ and S$if$ in figure 33 can be small switches as they connect only to the high-impedance current source and the gate of the OTA transistor. In addition, they operate at a fixed voltage of $V_{OTA}$, therefore they can be implemented as minimum-sized NMOS switches of $0.768/0.16\,\mu mm$.

Switches S$ic$ in figure 33 connect the top plate of the capacitor to the OTA output. These switches have to operate reliably over a wide voltage range, from $V_{OTA}$ to $V_{OTA} + V_{ref}$. If the voltage drop across the switches becomes too large, the OTA transistor will no longer be in saturation. Based on transient simulations, these switches were implemented as transmission gates with a $2/0.16\,\mu m$ PMOS and a $1/0.16\,\mu m$ NMOS.

The transmission gates S$ni$ and S$pi$ in figure 33, interconnecting the stages for operation in 10-bit mode are sized by considering the following trade-off: their on-resistance should be low enough for the combined sampling capacitances to act as one capacitance in terms of noise. However, too large switches will add large amounts of nonlinear capacitance to the sampling capacitors. The required switch resistance was determined by performing time-domain noise simulations of the VTC with a range of resistors interconnecting the sampling capacitances. These simulations will be discussed in detail in section 5.2.2. It was found that an on-resistance of about $500\,\Omega$ left sufficient noise performance in the system. This resulted in transmission gates with a $8/0.16\,\mu m$ PMOS and a $2/0.16\,\mu m$ NMOS.

### 4.2.2 Comparator

Design of the comparator was not executed within the time span of this assignment. However, it is a critical component of the topology: it has to resolve the point at which two slopes cross each other relatively slowly, with a timing accuracy below a gate delay, which corresponds to input-referred noise voltages below an LSB.

A straightforward implementation of the continuous-time comparator is expected to require quite some power, although it is not trivial to estimate this, since it depends on how much the bandwidth of the noise is limited at the cost of comparator speed [49]. However, certain specification points allow for a more tailored implementation. For one, the slopes always cross at approximately $V_{OTA}$, therefore the comparator does not need a wide common mode input range. Second, the comparator can be biased adaptively: an additional low-power comparator, for instance a slightly asymmetric differential pair, can detect when the slopes approach each other, and turn the main comparator on. After the decision has been made, the comparator can be switched off again.

Some other specifications to keep in mind is that the comparator input capacitance should not add too much nonlinear capacitance in parallel to the sampling capacitance, as this will impair the linearity of the converter. Finally, the comparator flicker noise can be an issue, as it causes an input-referred $V_{\text{offset}}$. Unlike the flicker noise in the current source, this offset is not compensated by the reference sampling mechanism, since this mechanism digitally divides $V_{\text{in}} + V_{\text{offset}}$ by $V_{\text{ref}} + V_{\text{offset}}$. Combining these two demands can be problematic, since a small input capacitance is achieved by small input devices, but a low flicker noise requires large input devices.

For simulation purposes, the comparator will be idealized as a limited voltage amplifier with a high gain.

## 4.3  Integration

This section treats the integration of the proposed TDC and VTC to form the overall system.

### 4.3.1  Synchronizer

As established in section 4.1.1, the fastest oscillation conditions of the ring oscillator, together with the number of flip-flops in the synchronizer for the 'start' signal, determine the time available to the VTC to perform the level shift after the sample has been taken. Until now the number of flip-flops was assumed to be two, resulting in 1 to 2 nanoseconds available to the VTC. However, early transient simulations showed that at least 5 nanoseconds are required for all nodes around the OTA to stabilize after the level shift. For this reason, the synchronizer is replaced by five flip-flops in the final circuit.

### 4.3.2  Control and timing logic

For correct operation of the system in both modes (10-bit single channel or 9-bit interleaved), some digital control logic is required. The logic consists of a simple state machine that accepts a single-ended clock of 11 MHz. The state machine is constructed such that the states change on the rising edge of the clock, and the states are used to control the corresponding actions on the falling edge of the clock.

In 10-bit mode, all channels track the input during one clock cycle, then perform the slope conversion for two clock cycles. In 9-bit mode, all channels track for one cycle, then perform slope conversion for one cycle. This requires a state machine that divides the input frequency either by two or three, which is done using three flip-flops and a multiplexer as shown in figure 37. The first flip-flop is pre-loaded, the others are pre-cleared. This results in a logical '1' cycling through either two or three flip-flops. The output of the first flip-flop tells all channels to sample on the next falling edge in 10-bit mode, but is used only for channel 1 and 3 in 9-bit mode. The output of the second flip-flop is used for channel 2 and 4 in 9-bit mode. The function of signals in 10-bit mode is written in red, the function of signals in 9-bit mode is written in blue.

To alternate between sampling the reference signal and sampling the input signal, the frequency of the 'sample' signal for each channel has to be divided by two again. This is done using two more state machines of two flip-flops, as shown in figure 38. In 10-bit mode, each channel should sample the same input at the next falling edge, in 9-bit mode, the channels should interleave. This is achieved by the four multiplexers.

The obtained 'sample next' signals change on the rising edge of the clock, and are not very accurate in time, as they depend on the propagation delay of several flip-flops. The structure depicted in figure 39 is proposed to obtain a 'sample' signal that changes on the falling edge of the clock. It shows a very short path from the clock to the critical sampling edge.

The path from the obtained 'sample' signals to the corresponding sampling switches is shown in figure 40. In 10-bit mode, all channels listen to the same 'sample' signal, whereas in 9-bit mode, each has its own signal.

After the sample switch has opened, level shifting has to be performed, by connecting the capacitor to the diode-connected OTA and lifting the shielding well. The sampling capacitor needs to be disconnected from the OTA again, before the next sample is taken. This non-overlapping behavior is accomplished by the structure shown in figure 41.
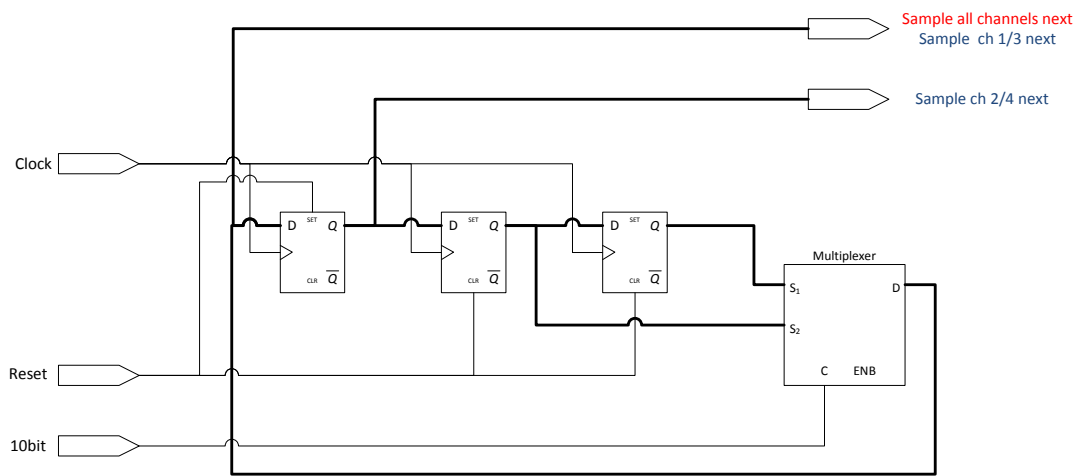
**Fig. 37:** State machine that provides a 'sample next' signal every 2 or 3 clock cycles, depending on the mode of operation.
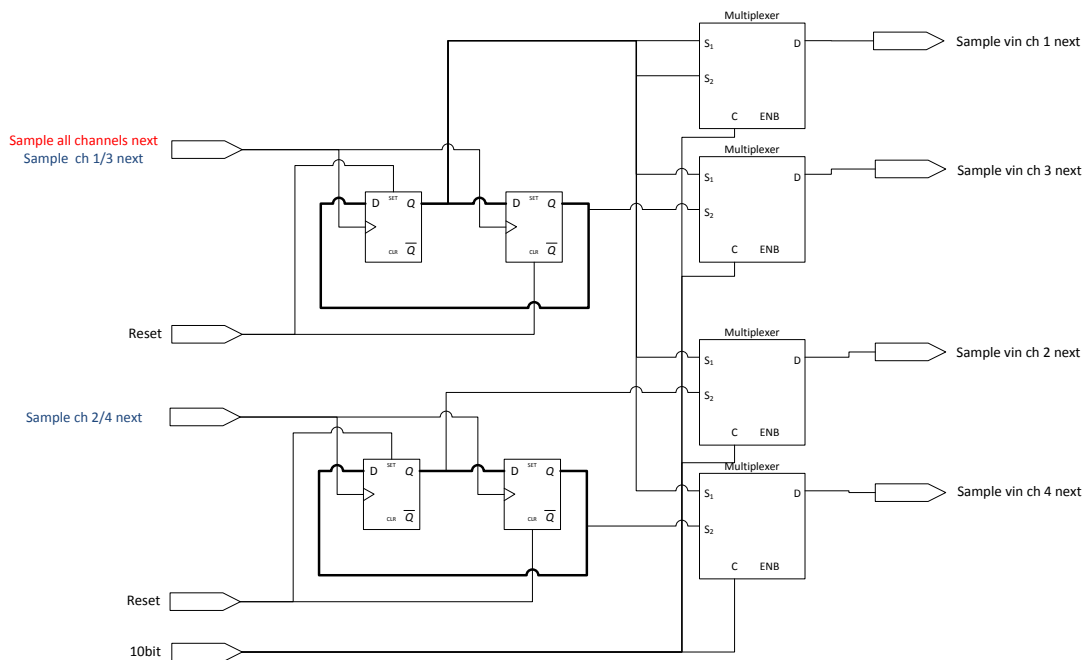
**Fig. 38:** State machine that divides the sample signals by 2 for alternating sampling of Vin and Vref. Depending on the mode of operation, the correct signal is routed to the individual channels.
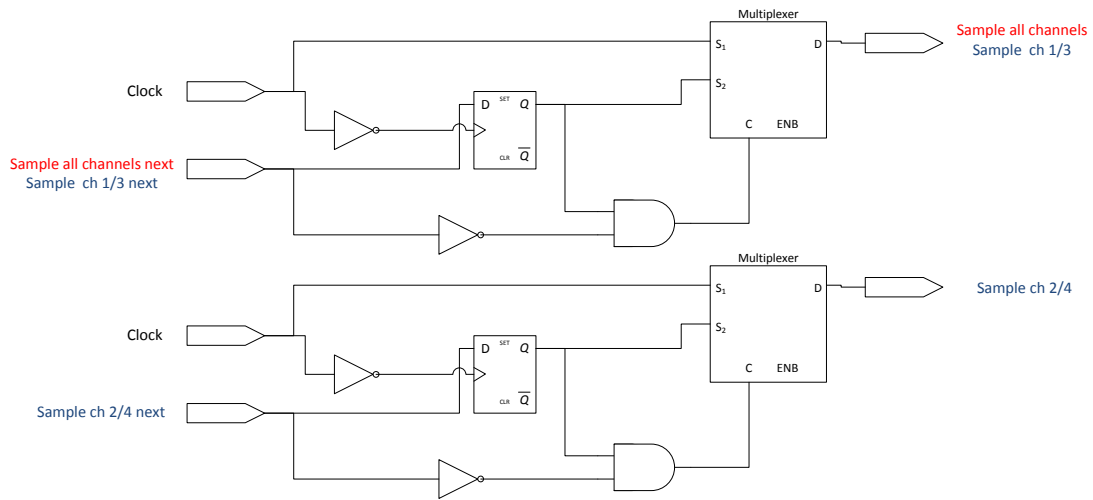
**Fig. 39:** Circuitry that uses the 'sample next' signals to generate a more accurate 'sample' signal on the falling edge of the clock.
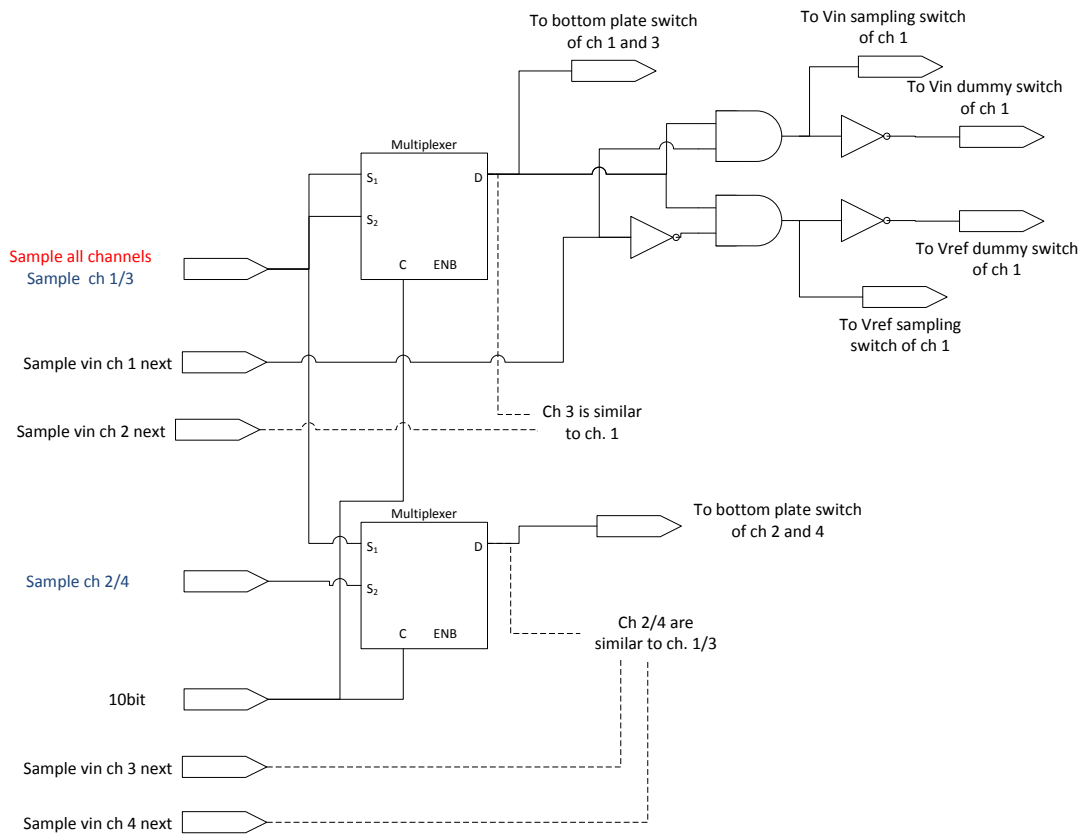


**Fig. 40:** Circuitry used to control the input and reference sampling switches and bottom plate switches, depending on the mode of operation.
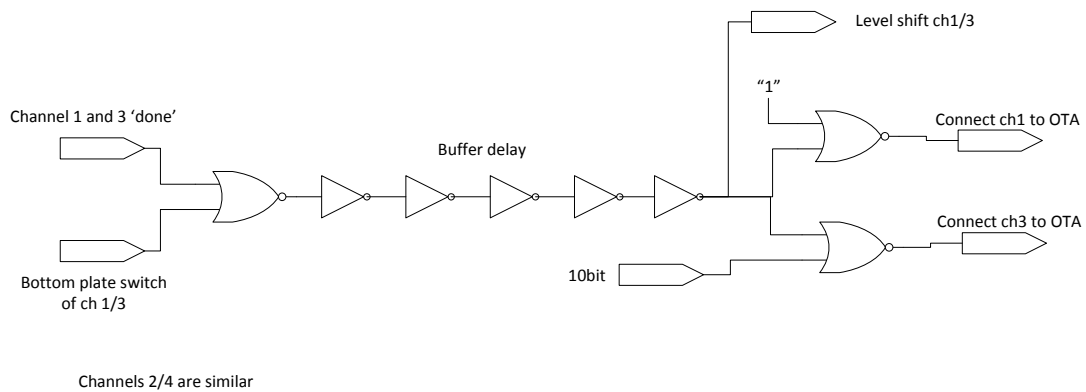
**Fig. 41:** Circuitry to control the level shift switches.

Now that level shifting is accomplished, the VTC must react to the 'start' signal from the TDC, by opening the diode-connected OTA and incorporating the capacitor in the loop. A short and equal path from the start signal to the switches is desired for each channel. Also, it is desirable to stop the slope after the conversion has finished, since otherwise the sampling capacitor is charged to the opposite polarity. This causes the top plate to be outside the supply rails when the bottom plate is connected to ground for the next sampling operation. The structure depicted in figure 42 accomplishes this.
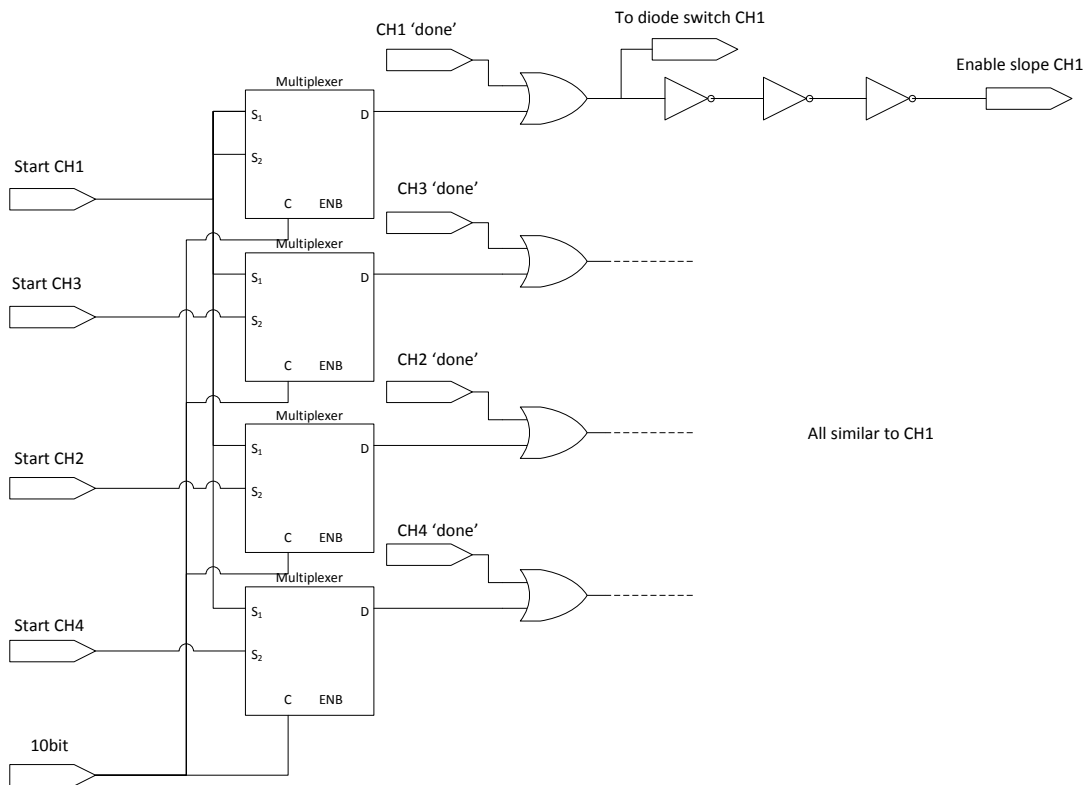


**Fig. 42:** Circuitry to control the switches that start the slope conversion.

And finally there is the issue of how to detect whether an individual channel is 'done'. Simulations have shown that simply using the comparator output to stop the channel results in an unstable situation: the channel will oscillate between stop and restart due to charge injection effects. Therefore, the comparator output is used to clock a flip-flop, to guarantee a non-ambiguous signal indicating if the channel is done.

All of the proposed digital circuitry was implemented on transistor level using gates from the logic library, and simulated; the results will be shown in the next chapter.
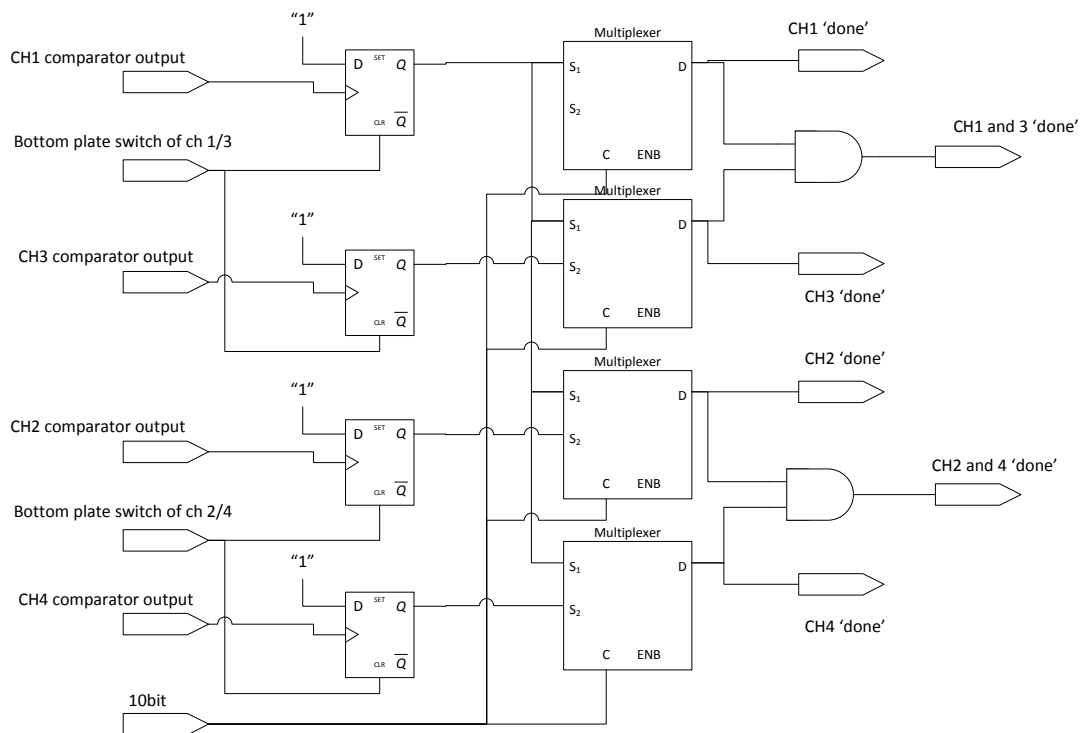


**Fig. 43:** Circuitry that detects if a channel is done.

## 4.4 System overview with implementation choices

During the implementation trajectory, several choices were made that can be reflected on system level, such as the number of stages in the oscillator, the number of VTC channels, the number of flip flops in the synchronizer and the number of bits in the coarse counters. Therefore figure 44 shows the system overview with these choices incorporated.

## 4.5 Conclusion on implementation

A transistor level implementation was proposed for the majority of the blocks in the system. It was seen to that the circuitry was kept highly digital and suitable for implementation in scaling CMOS technologies.

The TDC has taken the shape of a 16-stage, differential, free-running ring oscillator, optimized for phase noise. Its FoM is close to the theoretical limit for ring oscillators, indicating that little power was wasted in choosing a differential topology. It achieves a nominal delay of 90ps per stage. Digital flip-flops from the library were proposed to register its state for each VTC channel after testing for possible drawbacks. Transistor-level implementations were also proposed and simulated for decoding the fine bits and counting the coarse bits.

A reconfigurable VTC was implemented to act as either a single, 10-bit channel or four interleaved 9-bit channels. It was designed to achieve 12 bits of noise performance in the 10-bit mode, and by dividing the sampling capacitor into four equal parts, 11 bits of noise performance in the 9-bit mode. The system was deliberately designed with noise performance to spare, in order to better study the tonal and linearity behavior of a prototype. The system will be clocked with the same clock signal in both modes of operation. Transistor-level implementations of the control logic were devised and simulated.
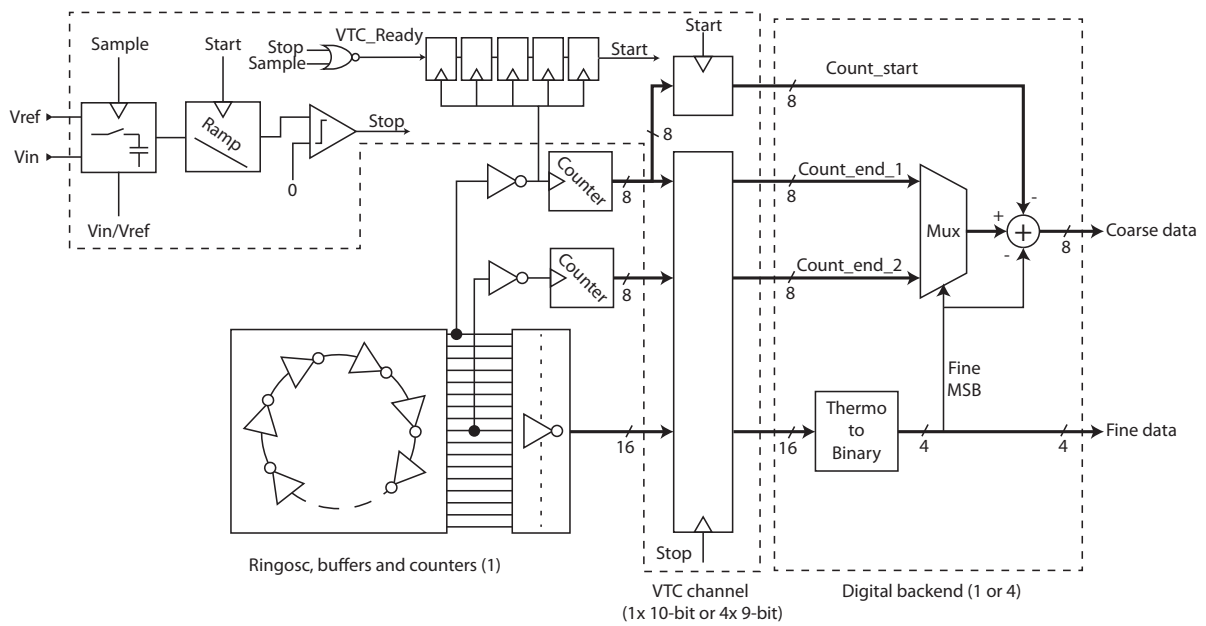
**Fig. 44:** System overview with choices from the implementation trajectory included.

The comparator, a non-trivial part of the VTC, was not implemented within the time span of this work. However, guidelines were given towards a power-efficient implementation tailored to this specific application. A major issue still to be conquered is flicker noise in the comparator, since it is not (fully) compensated on topology level by the reference sampling mechanism.

# 5   System performance

This section presents simulation results of the TDC and VTC and discusses their impact on the overall performance. It includes simulations of the combined system and concludes with some remarks on the results.

## 5.1   TDC

The functionality of the TDC was verified by transient simulations. The test structure included transistor level implementations of the oscillator, its output buffers, two counters on opposite phases and a bank of flip-flops for registering the state of the oscillator and counters. The flip-flops were clocked at varying points in time relative to the start of the simulation (imitating the 'stop' edge). The output values of the decoded fine bits and the LSBs of the two coarse counters were plotted for a range of arrival times of the 'stop' edge. The result is shown in figure 45.
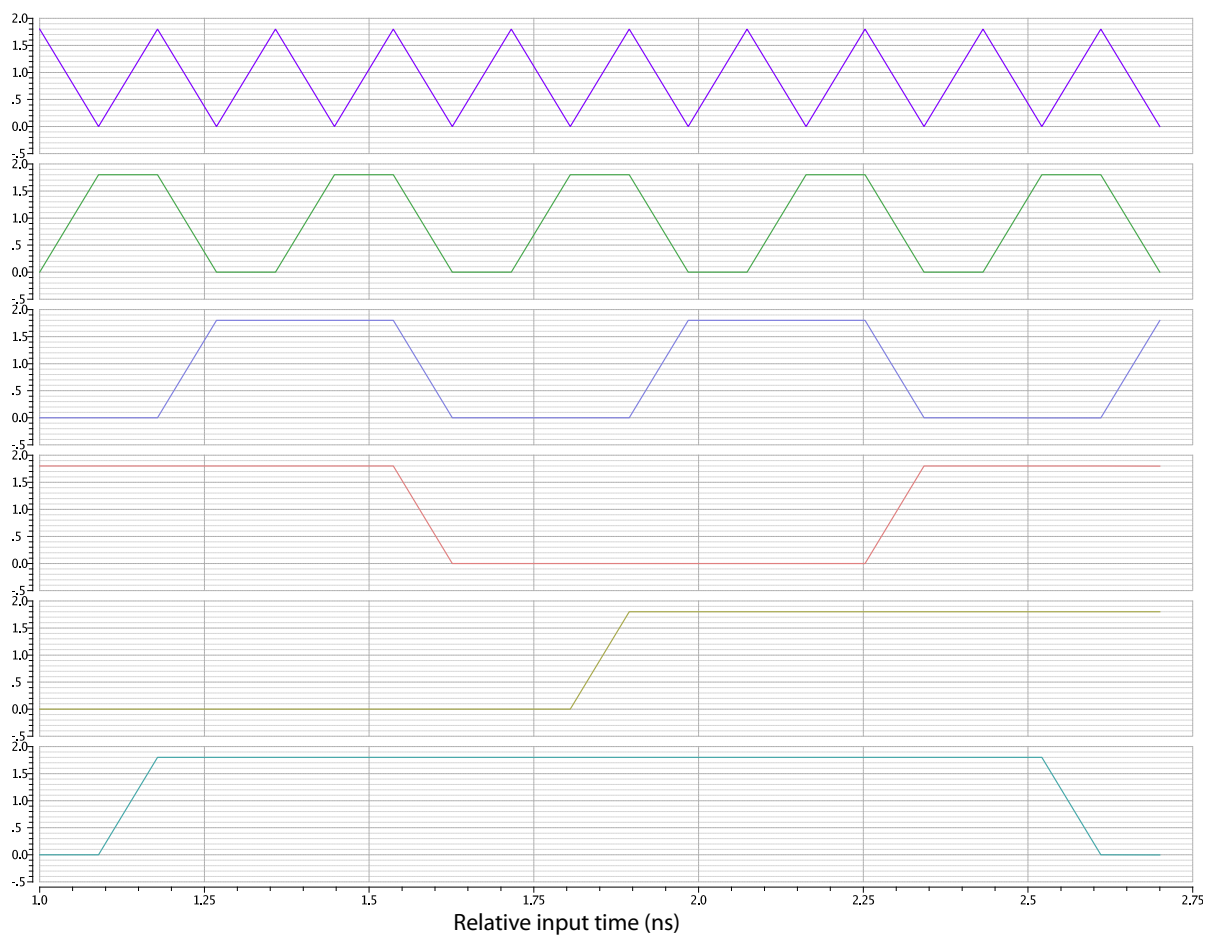


**Fig. 45:** Results of transient simulations of the TDC. The plot shows final values of the outputs for varying arrival time of the 'stop' signal. The signals shown are, from top to bottom: Bit 0 (LSB), 1, 2 and 3 (MSB) of the decoded oscillator state, and the LSBs of counter 1 and 2.

The apparent sawtooth shape of the signals is due to the limited amount of simulation points that are connected using straight lines. In reality, the final output values are digital: either high or low. The plots reveal a functional TDC: binary counting can be distinguished throughout the fine bits and on to the counter LSBs. Also, the toggling of the counter LSBs is placed correctly in between the toggling instants of the fine MSB. This is important, since it allows the fine code to correctly indicate which counter should provide a stable value at any time (refer to the metastability issue in section 3.4.1). If this were not the case, the coarse counters should be connected to a different oscillator output phase relative to the fine decoder.

Now that the functionality of the TDC is demonstrated, some key performance points can be summarized, including some that were already determined in the implementation chapter:

- Power consumption: On average, the oscillator consumes 857 $\mu$W, the buffers consume 160 $\mu$W (loaded by one flip flop on each phase) and the counters consume 232 $\mu$W each. The 16 flip-flops on the 16 output phases of the oscillator consume 390 $\mu$W in total.

- Linearity: The mismatch between stages was found to be 2.7% and will translate to a cyclic error pattern of 5.4% in magnitude (see section 4.1.1. The flip-flops are expected to add 3.5% of additional mismatch to the stages (section 4.1.2). Together, these small nonlinearities are not expected to dominate the linearity of the ADC, nor endanger its monotonicity.

- Thermal noise: The effects of thermal noise were calculated from the phase noise performance to lie a factor 35 below mismatch effects (section 4.1.1). This is theoretically sufficient for about 18 bits of dynamic range.

An interesting note is that the flip-flops and buffers consume much more power when their clock signal is low. This originates from the internal structure of the library flip-flops: these are flip-flops consisting of a master latch and a slave latch; a simplified version is shown in figure 46. When the clock signal is low, the master latch is constantly 'tracking' the data signal at high speed. Lots of power can be saved here by either designing custom flip-flops or by limiting the time the clock input of the flip-flops is low (e.g. by only lowering the clock when the slope approaches the zero-crossing).
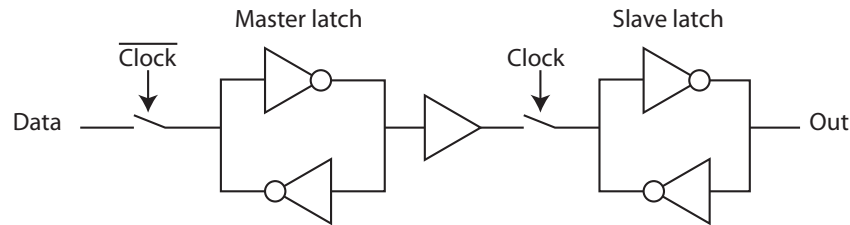


**Fig. 46:** Simplified schematic of the internals of a library flip-flop.

## 5.2   VTC

The voltage-to time converter was extensively tested. Its transient performance will be shown first, to provide a feel for the mechanisms. Next, noise simulations are presented and finally the linearity was put to the test.

### 5.2.1   Transient performance

Figure 47 shows some relevant signals during a transient analysis of the VTC during 10-bit operation. The signals show alternating sampling of the 1V reference and 500 mV input voltage for one period of the 11MHz clock, and a slope conversion during the two subsequent periods. The differential voltage across the sampling capacitor shows sufficiently fast settling to the input value, as expected from the dimensions of the sampling switches. The effectiveness of the assisted level shifting mechanism is also visible.

Some circuit imperfections are visible: there is a slight bump in the differential voltage after level conversion. This is due to charge injection effects and was found to be signal-independent. It will cause a slight offset in the conversion of both the input and the reference voltage, which may require further attention.

The common-mode voltage of the capacitor shows a small glitch as the diode-connected OTA is opened and the capacitor is included in the loop. This is also attributed to charge-injection effects into the high-ohmic node between the OTA gate and the current source. Some added decoupling on this node was found to suppress the glitch effectively, but further investigation is required.
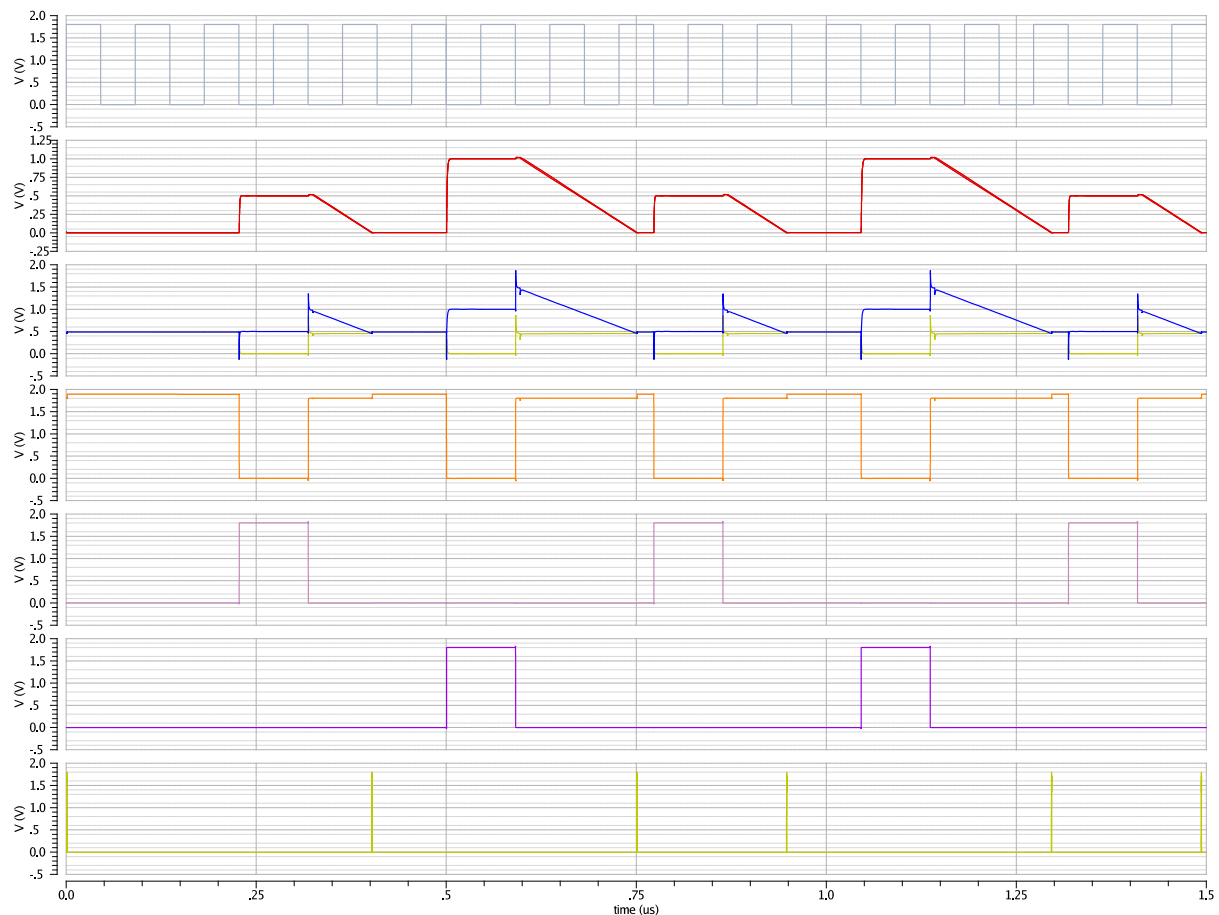


**Fig. 47:** Results of a transient simulation of the VTC in 10-bit mode, with a 500mV input voltage and a 1V reference voltage. From top to bottom: The 11 MHz clock signal; the differential voltage across the sampling capacitor; the voltage on the positive and negative terminals of the sampling capacitor; the voltage on the well of the sampling capacitor; the 'sample Vin' signal; the 'sample Vref' signal; the comparator output signal ('stop').

Figure 48 shows some relevant signals of a single channel in 9-bit operation. Although the sample clock is not shown, clearly an equal amount of time is now allotted for sampling and slope conversion. The glitches at the start of the slope conversion are also clearly visible in 9-bit mode.



**Fig. 48:** Results of a transient simulation of a single VTC channel in 9-bit mode, with a 500mV input voltage and a 1V reference voltage. From top to bottom: The 'Sample Vin' and 'Sample Vref' signals derived from the 11 MHz clock signal; the 'start' signal; the voltage on the positive and negative terminals of the sampling capacitor; the differential voltage across the sampling capacitor; the comparator output signal ('stop').

In figure 49, the differential voltages are depicted across all four sampling capacitors in 9-bit mode. The interleaved operation is clearly visible, producing an output value and a reference value each sample clock cycle.
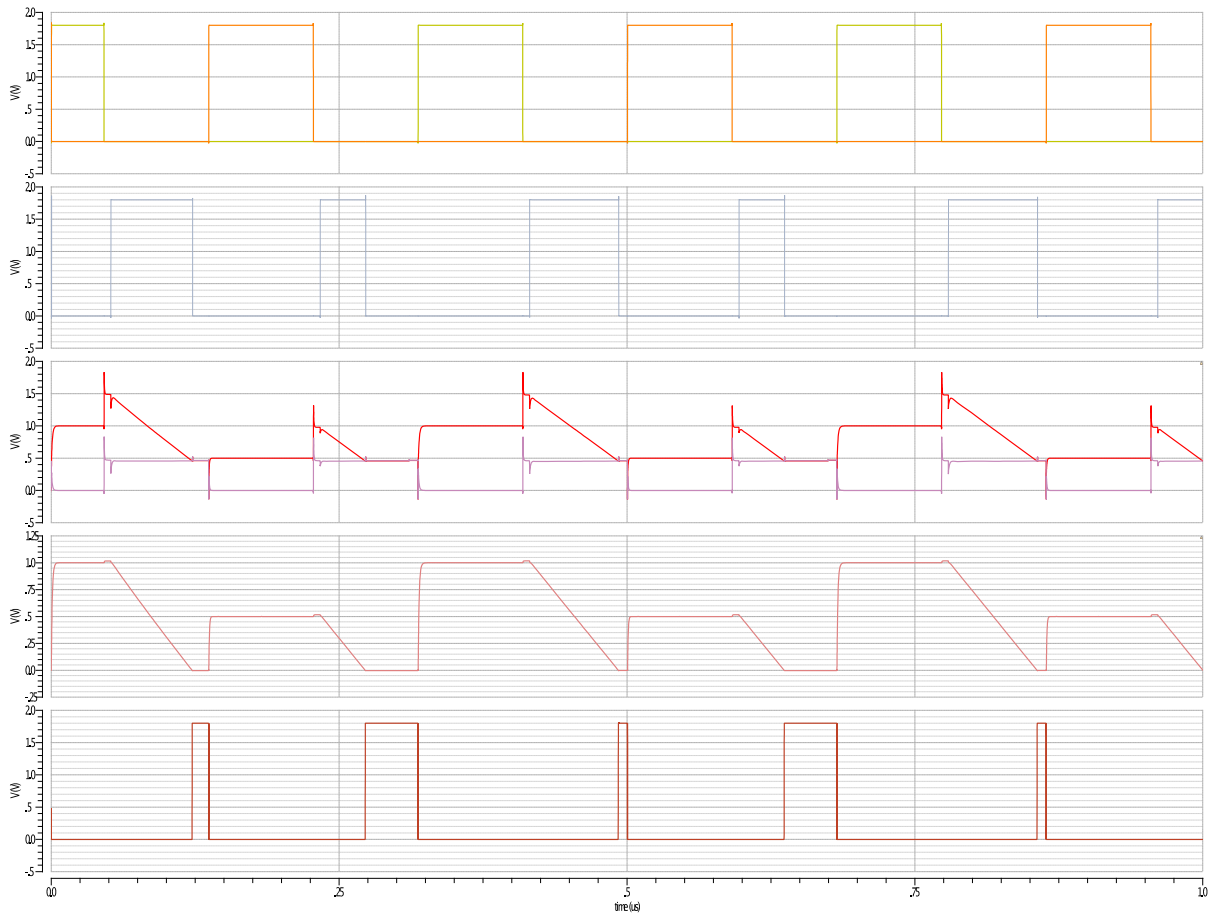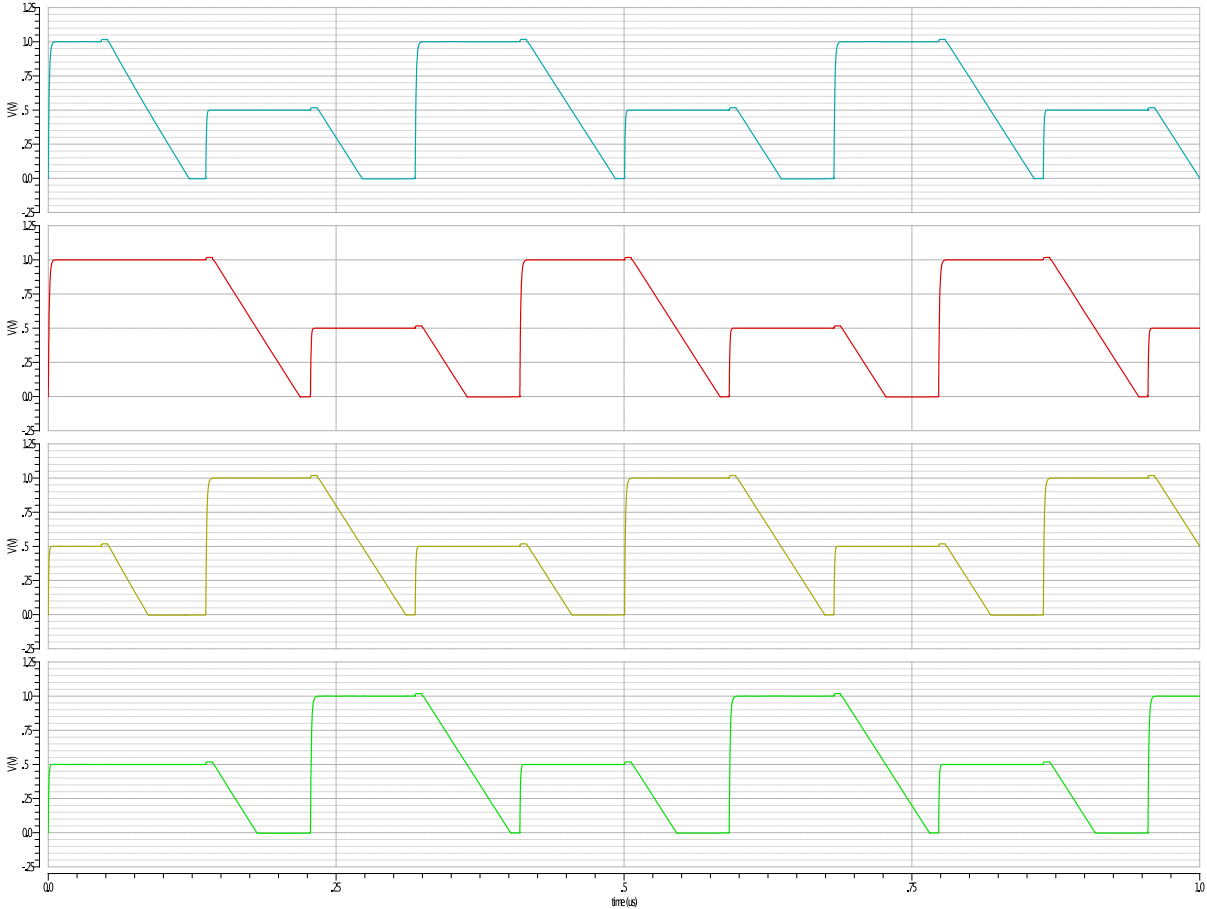
**Fig. 49:** Results of a transient simulation of all four VTC channels in 9-bit mode, with a 500mV input voltage and a 1V reference voltage. Shown are the differential voltages across the sampling capacitor for all four channels.

### 5.2.2  Noise performance

The VTC sampling capacitor and current source were implemented to theoretically achieve 12 bits of noise performance in 10-bit mode. By means of dividing the capacitance by four, 11 bits of noise performance should be obtained in 9-bit mode. However, these calculations assumed an ideal sampler, a resistive current source and ideal switches to interconnect the four sampling capacitors in 10-bit mode. This section shows the results of applying realistic components at these points in the VTC.

To study the impact of implementing the current source as a degenerated MOS current source and the switches as non-ideal switches, so-called "Time domain periodic noise analyses" were carried out using Spectre RF. The details of this type of analysis can be found in appendix D. It suffices to mention here that this analysis can be used to determine the voltage uncertainty between two nodes in a periodically operating circuit, at any time point during the period. This analysis accounts for noise folding or aliasing, and it can only be set up correctly if the bandwidth of the relevant noise sources is known. Details of how this analysis was set up for the problem at hand can also be found in appendix D.

Figure 50 shows the results of three time domain noise simulations. These simulations show the voltage uncertainty across the capacitor at various stages during the VTC operation, for a single channel (9-bit mode) converting a full-scale input voltage of 1V. The three curves were obtained using a resistive implementation of the current source, a degenerated MOS implementation of the current source with 1 nF decoupling on the bias line and a degenerated MOS with a more realistic 3.72 pF decoupling (i.e. a 20/20 $\mu$m MOS capacitor).
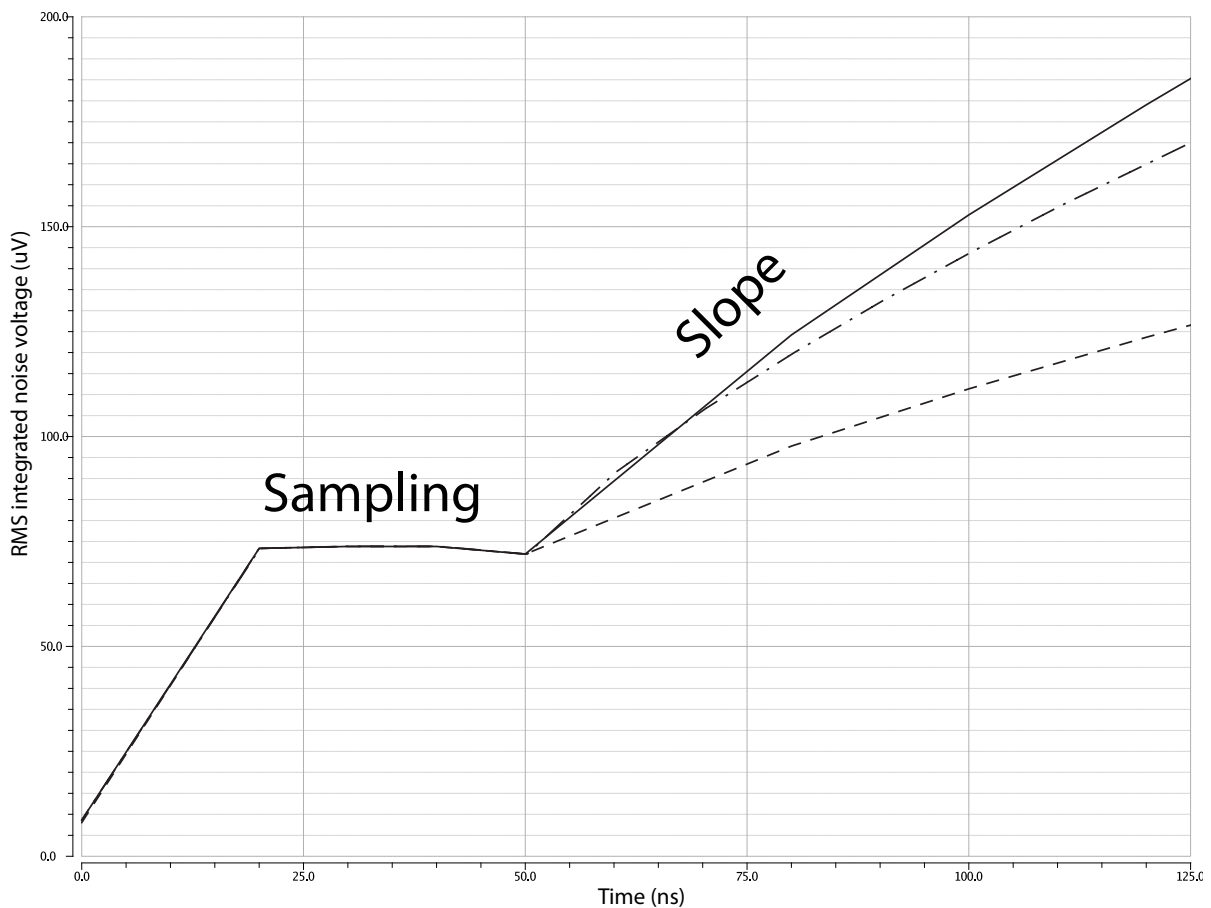


**Fig. 50:** Results of a time domain noise simulation of one VTC channel in 9-bit mode. Dashed line: current source is implemented as a poly resistor of 182 k$\Omega$. Dash-dot: current source is a degenerated PMOS implementation, with 1nF decoupling on the bias line. Line: current source is a degenerated PMOS implementation with a 20/20$\mu$m poly capacitor (3.72 pF) on the bias line.

All three implementations show an equal voltage uncertainty during and after sampling, but a different increase

in uncertainty while integrating the noisy current source, as expected. After sampling, a voltage uncertainty of $\sqrt{kT/C} = 86\mu\text{V}$ is expected for all three implementations. The simulations show about $74\mu\text{V}$. This deviation indicates that the simulation parameters were set too liberal to include all noise. This was necessary to maintain sufficient simulation speed to generate relatively detailed plots. Subsequent noise simulations were performed with greater accuracy, as also detailed in appendix D. In general, this resulted in figures within a few percent from the expected values. However, since these simulations are lengthy, only the two points of interest were calculated: The noise after sampling and the noise right before the zero-crossing of the slope.

Since the resistive implementation has approximately $V_{\text{dd}} - V_{\text{OTA}} = 1.33\text{V}$ headroom, a voltage uncertainty of $\sqrt{(kT/C) * (1 + 2V_{\text{in}}/V_{\text{cur}})} = 137\mu\text{V}$ is expected at the end of the ramp (refer to equation 10). The simulation shows $126\mu\text{V}$, again due to the fact that the simulation parameters were set too liberal.

The noise simulations on the degenerated MOS implementation of the current source in the previous chapter showed that it was about as noisy as a resistor with half the voltage headroom. Therefore an uncertainty of $\sqrt{(kT/C) * (1 + 2V_{\text{in}}/\frac{1}{2}V_{\text{cur}})} = 173\mu\text{V}$ is expected at the end of the slope. Simulations show $164\mu\text{V}$. More realistic decoupling of the bias line by 3.72 pF results in an additional $21\mu\text{V}$ of noise in the final value, coming from the reference branch.

In 9-bit mode with the MOS current source and realistic decoupling capacitor, the noise is $83\mu\text{V}$ after sampling, where $86\mu\text{V}$ is expected, indicating a good match between theory and simulation with the more accurate settings. The noise increases to $190\mu\text{V}$ at the end of the slope.

This type of simulation was also used to size the transmission gates that interconnect the channels in 10-bit mode. This resulted in transmission gates of (W/L)p = 8/0.16 and (W/L)n = 2/0.16. In 10-bit mode with the MOS current source and realistic decoupling, the noise amounts to $58\mu\text{V}$ before the slope, where $43\mu\text{V}$ was expected with ideal connections between the sampling capacitors. The noise increases to $120\mu\text{V}$ at the end of the slope.

When designing the VTC, the total noise at the end of the slope was designed to equal the quantization noise of an ideal quantizer of 12 bits for the 10-bit VTC and 11 bits for the 9-bit VTC. To see where the performance stands using non-ideal components, the comparison can be made again. The input-referred quantization noise of an ideal quantizer is given by [42]:

$$\sigma_v = \sqrt{\frac{V_{\text{in,peak}}}{12 \cdot 2^{\text{B}}}} \tag{11}$$

with B the number of bits. Therefore, 12 bit noise performance equals $70\mu\text{V}$ of noise, 11 bit equals $141\mu\text{V}$, 10 bit equals $282\mu\text{V}$ and so on. So all noise sources included, the VTC shows better than 11 bit noise performance in 10-bit mode and better than 10-bit noise performance in 9-bit mode. Therefore the converter will still be quantization noise limited, which was intended in order to better study its tonal and linearity performance. Furthermore, there is margin for some added noise from the comparator.

### 5.2.3 Linearity

To determine if the VTC circuit is sufficiently linear in both modes, the input voltage was swept from 0V to 1V in steps of 20 mV and the zero-crossing instant of the comparator was logged.

Figure 51 shows the zero-crossing time as a function of the input voltage, as well as its derivative (the voltage-to-time gain for varying input voltage). The small disturbances are most probably simulation inaccuracies. A slope is visible in the derivative of the transfer function, indicating mainly second-order harmonic distortion (HD2). This is expected, since the system was constructed in a single-ended fashion, and there is always some A rough, 3-point linearity calculation along the lines of [50] revealed a HD2 component at -68 dB and a negligible HD3 component. Comparing this to the SNR of an ideal 9-bit quantizer $SNR = 6.02 \cdot N_{\text{bits}} + 1.76 = 56\text{dB}$ reveals that this HD2 performance is acceptable for a practical ADC. There is still some room for added nonlinear capacitance to the system, such as parasitics and the input capacitance of a realistic comparator.



**Fig. 51:** Output time as a function of input voltage of the VTC in 9-bit mode (top) and its derivative for linearity calculations (bottom).

Figure 52 shows the same plots for the system in 10-bit mode. A remarkable difference with the 9-bit mode is that a parabola-shaped derivative is now visible, indicating a substantial amount of third-order harmonic distortion (HD3). Indeed, the same type of linearity calculation now shows an HD2 of -76dB and an HD3 of -60dB. The SNR of an ideal 10-bit quantizer is 62 dB, so this performance is barely acceptable for a 10-bit ADC. The third-order nonlinearity is probably due to the nonlinear capacitances added by the rather large transmission gates that interconnect the sampling capacitances. This requires further investigation.

Both situations also show that the range for very low input voltages may be unusable. This is caused by the comparator having to toggle almost immediately after the slope has started. The signals that have led to the discharging of the capacitor are then mostly glitches caused by the switching of the OTA. This topic also requires further attention.
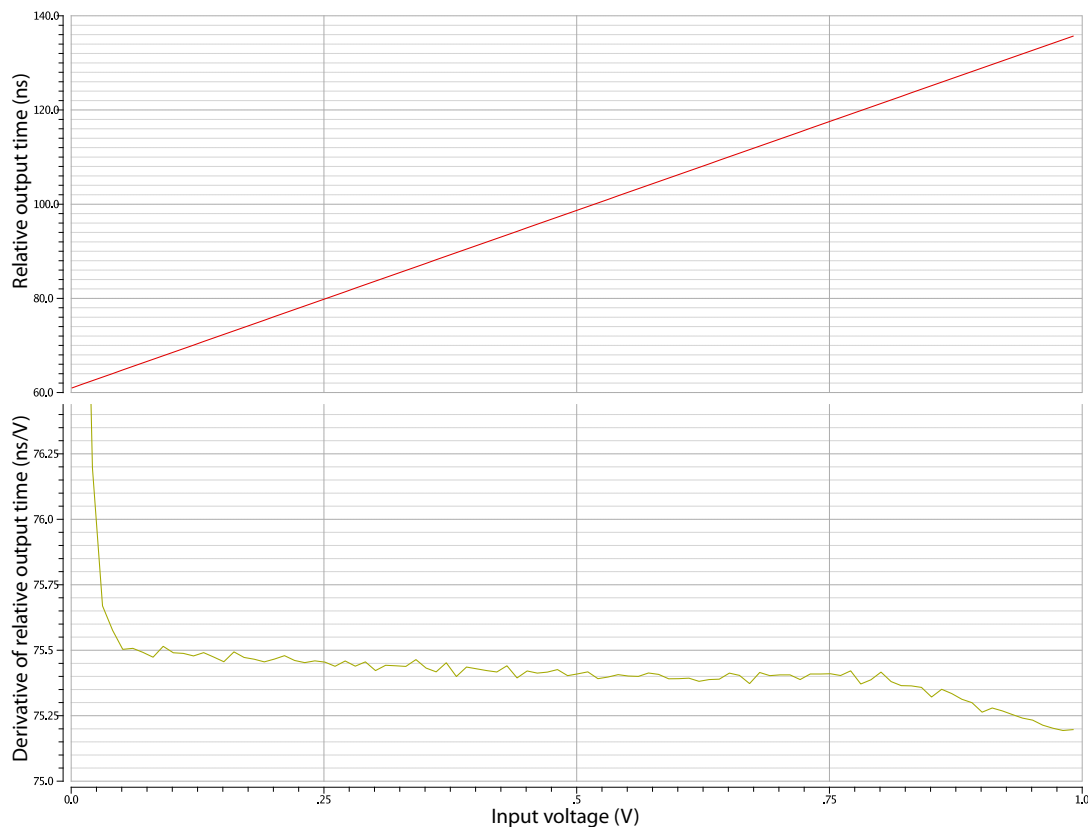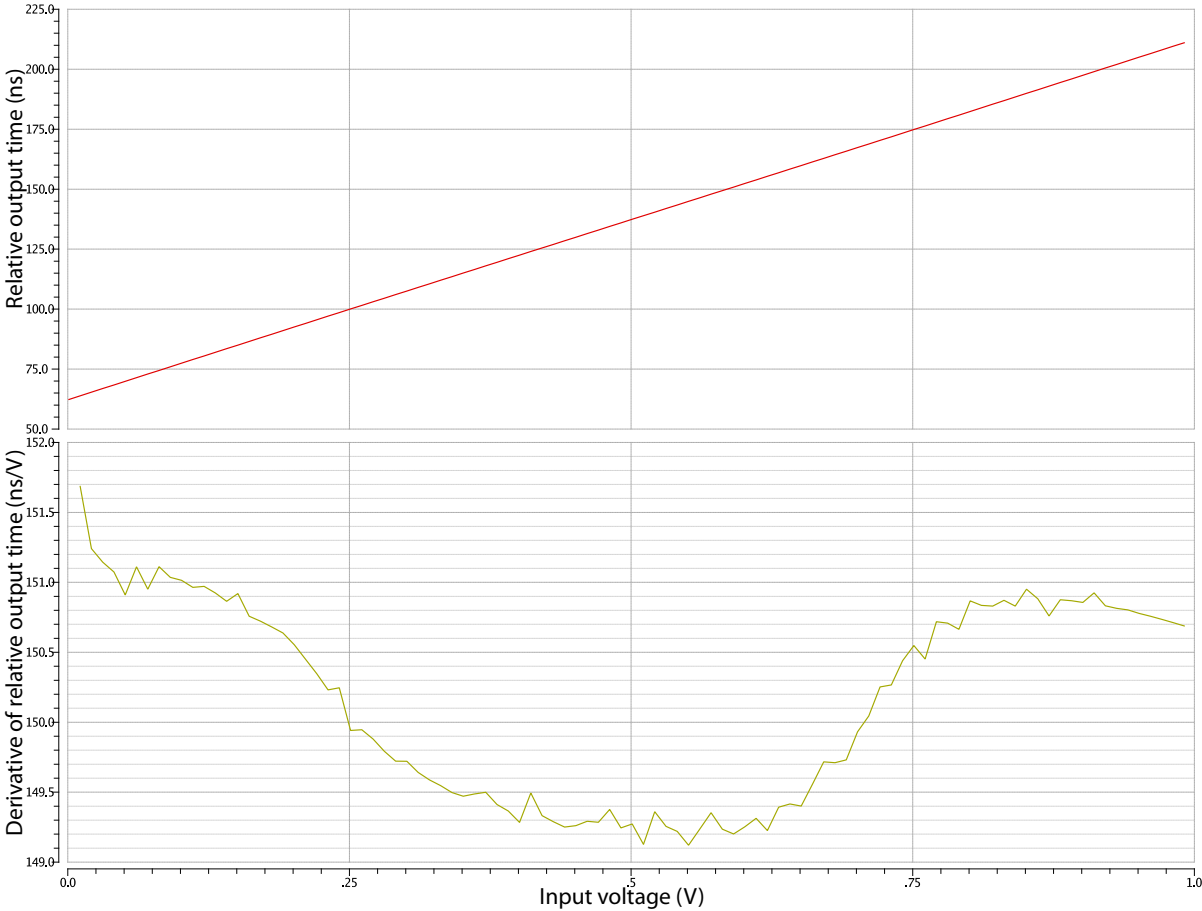
**Fig. 52:** Output time as a function of input voltage of the VTC in 10-bit mode (top) and its derivative for linearity calculations (bottom).

## 5.3   Complete system

Figure 53 shows a transient simulation of the complete system, configured in 10-bit mode, operating from the 11 MHz sample clock. The reset signal for the VTC is resynchronized with the falling edge of the clock, such that the state machines start operating on the next rising edge. One phase of the asynchronous ring oscillator output is shown, as well as the LSBs of the two counters on opposite phases.

The analog VTC voltages demonstrate the effectiveness of the 5-flip-flop synchronizer used between the 'VTC_ready' and 'Start' signals: almost 10 ns is allowed for level shifting.
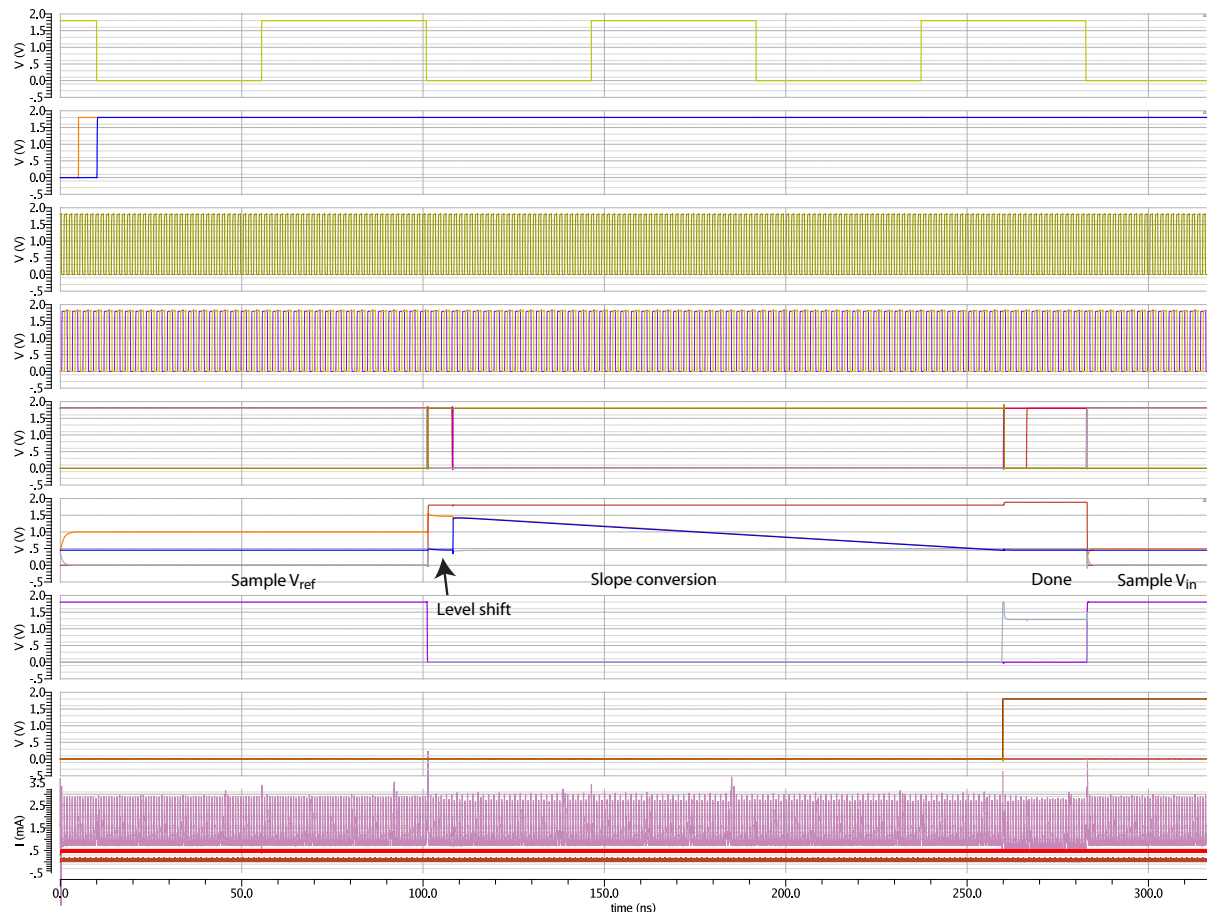


**Fig. 53:** Transient simulation of the complete system in 10-bit mode. From top to bottom: the 11 MHz clock signal; the reset signal for the VTC and its synchronized version on the falling edge of the clock; one phase of oscillator output; the LSBs of the two anti-phase counters; several digital control signals inside the VTC; the analog voltages inside the VTC; the 'start' signal (active low) and the comparator output ('stop' signal); the coarse and fine bit signals assuming a value at the stop signal; the power consumption of the complete system (purple), oscillator (red) and output buffers (orange).

**Power consumption**

Rough estimates of the power consumption can be made from the results. The overall converter draws 2.18 mW. The main contribution of the TDC in this consumption consists of the oscillator consumption ($817\mu$W), the output buffers ($160\mu$W), the counters ($2x232\mu$W) and the flip-flops ($396\mu$W). This leaves roughly $300\mu$W associated with the VTC. Since the VTC core consumes only the two bias currents of $2x14.7\mu$A = $52\mu$W, the consumption is dominated by its control and timing circuitry.

The FoM of the converter can be estimated: Since three 11MHz clock cycles are needed to produce an output sample, and the reference sampling mechanism also requires three cycles, the effective sample rate is 1.83 MHz.

Applying the Walden FoM [51]:

$$\text{FoM} = \frac{P}{F_\text{s} \cdot 2^B} \tag{12}$$

results in a FoM of 2.32 pJ/conversion-step. This is quite far from state-of-the art [52] considering it excludes the power consumption of the comparator and the external logic required to perform the digital division. However, the power consumption is dominated by digital circuitry, so huge improvements are expected when this topology is ported to a newer technology, as detailed in section 5.4.

Besides this, the real strength of the topology lies in the interleaved operation of multiple VTC channels on one time base. Unfortunately, time did not permit simulating the entire system in 9-bit mode, however, a performance estimate can be made. The consumption of the VTC core increases to 79 μW caused by the added current sources. The control logic is expected to draw approximately the same amount of power. The oscillator core consumption will remain unchanged, the power consumption of its output buffers will increase in order to drive four channels of flip-flops. An increase of about factor two is estimated. Four banks of flip-flops will consume four times the 396 μW of their current configuration. The resulting power consumption of the system will be about 3.61 mW. In 9-bit mode, the interleaved system will produce an output sample every clock cycle of 11 MHz. This results in a FoM of 640 fJ/conversion-step. Again, this is not state-of-the-art, but promising for such a highly digital architecture.

## 5.4   Performance estimate in 65 nm CMOS

In order to estimate the performance improvement with technology, the oscillator and its output buffers were implemented in 65nm CMOS. Following the same approach as that of section 4.1.1, the transistors were sized based on the internals of digital library gates. For NMOS transistors of 0.2/0.06 μm and PMOS transistors of 0.28/0.06 μm, the nominal oscillation frequency of the oscillator was found to be 807 MHz. The power consumption of the oscillator was drastically reduced compared to 140nm CMOS, at 62 μW from the allowed 1.2V supply. The buffers consumed 10 μW. For both the oscillator and buffers, this is more than an order of magnitude less power consumption per propagation delay than demonstrated in section 4.1.1 for 140 nm CMOS. Assuming this reduction is representative for the reduction in the other digital blocks, this is expected to result in about an order of magnitude better FoM of the overall converter in 65nm CMOS, making it approach state-of-the-art performance.

## 5.5   Conclusion on performance

The functionality of the VTC, TDC and combined system was largely demonstrated through various simulations.

The TDC was verified to function correctly by transient simulations. Combined with the results from the implementation chapter on its linearity, power consumption and noise properties, its overall performance is adequate for the application.

The functionality of the VTC was also verified in both modes by transient simulations. Furthermore, time domain noise simulations agree with the calculated values and show that the system has noise margin left for the inclusion of a non-ideal comparator. Linearity simulations indicate that the system needs some work, showing an undesired HD3 component in 10-bit mode. This is attributed to the nonlinear capacitance added by the transmission gates that interconnect the channels and requires further attention. However, only about 3 dB of linearity needs to be gained for acceptable performance.

Transient simulations of the complete system in 10-bit mode allowed for a rough estimation of its power consumption, which resulted in a figure-of-merit of 2.32 pJ/conversion-step in 10-bit mode and, by extrapolating the results from the test in 10-bit mode, 640 fJ/conversion-step in 9-bit mode. Both are quite far from state-of-the-art and still exclude the comparator. However, most of the power is consumed by digital blocks, which shows there is room for significant improvement in a more modern CMOS process. Preliminary simulations indicate that in 65nm CMOS, the performance of this concept can indeed compete with state-of-the-art converters.

There are some points that require attention, such as the glitches and other charge injection effects in the VTC. Also, the power consumption of the flip-flops may be reduced significantly with relatively little effort.

## 6   Discussion of portability, scaleability and limitations of the concept

Reflecting on the results so far, some qualitative remarks can be made on the scaleability, technology-portability and limitations of the concept.

### 6.1   Technology-portability

The circuit inventory shows many digital blocks, which are by definition technology-portable. The more critical components are the oscillator and the VTC.

As for the oscillator, the proposed implementation is expected to survive technology scaling, as each stage consists of a stack of two NMOS transistors and two PMOS transistors, which is also about the minimum 'stack' that is required to implement some functional digital gates. An expected issue with scaling transistors is that the flicker noise corner in the oscillator moves upward in frequency, necessitating more frequent reference conversions. If this becomes a real problem, it may be necessary to switch to a simpler inverter-based ring oscillator with an odd number of stages at the cost of some additional digital hardware, since in such oscillators, flicker noise can be effectively suppressed by designing for equal rise and fall times [47].

The VTC was also designed to be compatible with newer technologies: the OTA transistor in the VTC consumes little more than one threshold voltage from the available supply, leaving room for a large swing across the capacitor on one side, and a large headroom to implement the current source on the other side. This allows a large input voltage swing, which is in turn beneficial for the requirements of the comparator and for the size of the capacitor that is required for sufficient noise performance. A large headroom for the current increases its effective noise resistance and therefore also reduces the required capacitor size. Nonetheless, as technology scales, the main issue to be expected from the VTC is that the sampling capacitor becomes very large due to the limited headroom for both the input voltage swing and the current source. Also, there may be insufficient headroom to degenerate the current source to increase its effective noise resistance. Another issue that can be expected in the VTC is that a bootstrap circuit will become necessary for sufficient linearity of the sampler switch.

### 6.2   Scaleablilty

As for the scaleability of the system, the main question is how far the concept can be pushed in terms of speed and number of bits.

As demonstrated in section 4.1.1, the cyclic property of the mismatch and the low thermal noise of inverter-based delay stages theoretically allow for the calculation of many bits in the TDC. However, there is a more practical upper limit to the number of bits, since the calculation of more bits requires more time. During such lengthy conversions, low-frequency effects in the TDC come into play, such as supply voltage drift, temperature drift and flicker noise in the oscillator. These effects can only be compensated if a reference conversion is performed at sufficiently regular intervals, which is not possible anymore when the conversions become more lengthy than the time constant of the degrading effects.

Also, with increasing resolution, the sampling capacitance will increase by a factor four for each bit. This becomes impractical at some point (e.g. a 13-bit version of the current implementation, including 2 bits of over-design, would require about 150 pF of sampling capacitance). Besides, while the capacitance is increased by a factor four, the TDC requires only twice the time for conversion. Discharging four times the capacitance in twice the time requires twice the current. As a result, power consumption of the VTC increases by a factor of two for each added bit and may quickly dominate the power consumption of the converter, especially in more modern technologies where the digital blocks consume little power.

In addition, for each added bit, the slope on the capacitor voltage becomes less steep. The comparator must resolve when this slower ramp crosses zero, but still with gate-delay accuracy. This requires more gain and less input-referred noise voltage from the comparator and therefore more power.

Aside from noise and other unpredictable effects, the effective number of bits also limited by the linearity performance. From the results chapter, the limiting factor turned out to be the VTC already for 10-bit resolution. This excludes any nonlinear capacitance that will be added to the sampling capacitor by a realistic comparator. So if

a higher number of bits is desired, the VTC architecture needs work; for instance, it may be possible to devise a differential architecture based on the same principles.

The lower end of the useful number of bits is determined partly by the timing overhead: in the present concept, some time is required to resynchronize the 'VTC_ready' signal to the start of the conversion. During this time, level conversion takes place in the VTC. If, after level conversion, only a few bits need to be calculated, this timing overhead becomes significant. For example, from 6 bits downward, the actual conversion is expected to take about as much time as the time required for level shifting.

Since the linearity of the converter is also more relaxed for a low number of bits, completely different design choices can be made. For instance, a flash TDC may be preferred over a flash ring TDC and a simpler VTC architecture can be used that does not require level shifting and a virtual ground node. Even VCO-based architectures may be the preferred choice.

## 6.3 Conclusion

In conclusion, the proposed topology lends itself best for implementing a highly digital ADC of moderate resolution: higher than that achievable with a VCO-based ADC due to its superior linearity, but limited at the upper end by the size of the sampling capacitance becoming impractical and the conversions becoming so lengthy that they suffer from low-frequency effects. In general, the concept is expected to be highly portable with scaling technology.

# 7   Conclusion

In this work, an uncommon way of analog-to-digital conversion was investigated. The conversion takes place by first converting the analog input signal to a time difference between two digital transitions using a voltage-to-time converter (VTC), and subsequently digitizing this time difference using a time-to-digital converter (TDC).

Since TDCs can be highly digital structures, key performance points of the converter, such as power consumption, speed and silicon area, are expected to improve inherently with scaling CMOS technology. Due to the digital design flow, a high ease of implementation was also expected. Furthermore, a form of reconfigurability could be exploited that is not trivial in conventional ADCs: trading conversion time for accuracy. The assignment was to design and implement such an analog-to-time-to-digital converter and explore these possible advantages in the process.

An extensive study of many TDC topologies from literature was performed, the most important conclusion being to keep things simple if the application does not put any specific requirements on the TDC. Ring structures were found to be especially interesting, because of their good linearity properties. However, most simple TDC topologies are fundamentally limited in power consumption because the toggling of one digital gate is required for the calculation of each quantization step.

A system architecture was proposed in which the TDC was implemented as a 'flash ring' TDC, more specifically an asynchronous ring oscillator with flip-flops to record its state. To break the fundamentally limited power consumption of this flash TDC, it was noted that multiple VTC channels can make interleaved use of this TDC.

Possible topologies for the VTC were investigated more conceptually and superficially. To benefit from the superior linearity of the ring TDC, the VTC was implemented as a start-voltage controlled single-slope converter. A reference sampling mechanism was included in order to digitally compensate for process, voltage and temperature variations, as well as several flicker noise sources.

The TDC and VTC were largely implemented on transistor level in 140 nm CMOS. The TDC was designed for sufficient phase noise performance and functionally tested using transient simulation. The VTC was implemented to function as either one 10-bit channel or four interleaved 9-bit channels. Its functionality and noise performance were found to match the theory, and to be sufficient in both modes. Its linearity performance in the 10-bit mode remains a point of attention. Implementing the continuous-time comparator required within the VTC was not performed within the time span of the thesis.

A simulation of the entire system in 10-bit mode demonstrated its functionality and allowed for a rough figure-of-merit calculation of 2.32 pJ/conversion-step, which is expected to reduce to 640 fJ/conversion-step in interleaved 9-bit mode. In both cases, the power consumption is dominated by digital blocks, therefore the system is expected to achieve much improved performance in more modern CMOS technologies. Preliminary simulations in more modern 65nm CMOS show that state-of-the-art performance can be expected in this case.

In conclusion, the proposed converter still requires some work and will not achieve state-of-the-art performance in 140 nm CMOS, but it does demonstrate all the expected advantages, which makes it a promising concept for analog-to-digital conversion in scaling CMOS technology.

## 7.1   Recommendations

- The concept has potential for state-of-the-art performance, since its performance is dominated by the power consumption of digital blocks. However, it will not achieve state-of-the-art performance in CMOS14. Careful thought should be given if it is worthwhile to carry out a tape-out in CMOS14.

- The implications of digitally dividing the converted input voltage by the converted reference voltage should be investigated. If the effects are unacceptable, a revision of the VTC architecture may be in order, such that a full scale input voltage inherently maps onto a full scale TDC output.

- The oscillator was designed from a relatively arbitrary starting point of transistor sizes found in digital gates. It could be useful to investigate if this was the right choice or if larger transistors result in a more useful design point in terms of speed, power and phase noise.

- The assisted level switching mechanism applied in the VTC requires attention from someone more experienced in these mechanisms. It is suspected that substrate-well parasitics will play a significant role in the functionality of this circuit.

- Several imperfections in the VTC, mostly caused by charge injection effects, require further attention.

- The comparator is yet to be implemented, some guidelines were given as to how it can be tailored to the application. Flicker noise from the comparator is expected to form a problem, since unlike other flicker noise sources, its effects will not be fully compensated by the reference sampling mechanism.

- Power-hungry components such as the comparator, but also the flip-flops, need only be enabled close to the threshold crossing instant of the slope. It is advised to use a low-power, slightly asymmetrical differential pair to detect whether the zero-crossing of the slope is near. The resulting signal can be used to enable the comparator and the flip-flops only when necessary. If the consumption of the flip-flops cannot be reduced sufficiently using this technique, consider designing custom flip-flops.

## References

[1] Robert B. Staszewski. *Digital deep-submicron CMOS frequency synthesis for RF wireless applications.* PhD thesis, University of Dallas, 2002.

[2] Harlan W. Lefevre and James T. Russell. Vernier chronotron. *Rev Sci Instrum*, 30(3):159–166, 1959.

[3] J. Christiansen. Picosecond stopwatches: The evolution of time-to-digital converters. *IEEE Solid-State Circuits Magazine*, 4(3):55–59, 2012.

[4] Minjae Lee, M. E. Heidari, and A. A. Abidi. A low-noise wideband digital phase-locked loop based on a coarse–fine time-to-digital converter with subpicosecond resolution. *IEEE Journal of Solid-State Circuits*, 44 (10):2808–2816, 2009.

[5] M. S.-W. Chen, D. Su, and S. Mehta. A calibration-free 800 mhz fractional-n digital pll with embedded tdc. *IEEE Journal of Solid-State Circuits*, 45(12):2819–2827, 2010.

[6] Chun-Ming Hsu, M. Z. Straayer, and M. H. Perrott. A low-noise wide-bw 3.6-ghz digital delta-sigma fractional-n frequency synthesizer with a noise-shaping time-to-digital converter and quantization noise cancellation. *IEEE Journal of Solid-State Circuits*, 43(12):2776–2786, 2008.

[7] R. B. Staszewski, S. Vemulapalli, P. Vallur, J. Wallberg, and P. T. Balsara. 1.3 v 20 ps time-to-digital converter for frequency synthesis in 90-nm cmos. *IEEE Transactions on Circuits and Systems—Part II: Express Briefs*, 53(3):220–224, 2006.

[8] J. Borremans, K. Vengattaramane, V. Giannini, and J. Craninckx. A 86mhz-to-12ghz digital-intensive phase-modulated fractional-n pll using a 15pj/shot 5ps tdc in 40nm digital cmos. In *Proc. IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC)*, pages 480–481, 2010.

[9] J. Borremans, K. Vengattaramane, V. Giannini, B. Debaillie, W. Van Thillo, and J. Craninckx. A 86 mhz–12 ghz digital-intensive pll for software-defined radios, using a 6 fj/step tdc in 40 nm digital cmos. *IEEE Journal of Solid-State Circuits*, 45(10):2116–2129, 2010.

[10] V. Kratyuk, P. K. Hanumolu, K. Ok, Un-Ku Moon, and K. Mayaram. A digital pll with a stochastic time-to-digital converter. *IEEE Transactions on Circuits and Systems—Part I: Regular Papers*, 56(8):1612–1621, 2009.

[11] Seon-Kyoo Lee, Young-Hun Seo, Yunjae Suh, Hong-June Park, and Jae-Yoon Sim. A 1ghz adpll with a 1.25ps minimum-resolution sub-exponent tdc in 0.18um cmos. In *Proc. IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC)*, pages 482–483, 2010.

[12] T. Hashimoto, H. Yamazaki, A. Muramatsu, T. Sato, and A. Inoue. Time-to-digital converter with vernier delay mismatch compensation for high resolution on-die clock jitter measurement. In *Proc. IEEE Symp. VLSI Circuits*, pages 166–167, 2008.

[13] E. Raisanen-Ruotsalainen, T. Rahkonen, and J. Kostamovaara. An integrated time-to-digital converter with 30-ps single-shot precision. *IEEE Journal of Solid-State Circuits*, 35(10):1507–1510, 2000.

[14] Joakim Bergs. Design of a vco based adc in a 180 nm cmos process for use in positron emission tomography. Master's thesis, Lule University of Technology, 2010.

[15] S. Masuda, T. Watanabe, S. Yamauchi, and T. Terasawa. All-digital quadrature detection with tad for radio-controlled clocks/watches. *IEEE Transactions on Circuits and Systems—Part I: Regular Papers*, 56(2):285–293, 2009.

[16] T. Watanabe, S. Masuda, and H. Wakairo. An rcc receiver ic with tad-dqd and adpll using frequency multiplying number with decimals. In *Proc. Joint with the 22nd European Frequency and Time forum Frequency Control Symp. IEEE Int*, pages 478–481, 2009.

[17] T. Watanabe, S. Yamauchi, and T. Terasawa. A 0.0027-mm2 9.5-bit 50-ms/s all-digital a/d converter tad in 65-nm digital cmos. In *Proc. 16th IEEE Int. Conf. Electronics, Circuits, and Systems ICECS 2009*, pages 271–274, 2009.

[18] R.H.M. van Veldhoven. Analog to digital conversion circuit and method, Jun 2011.

[19] T. E. Rahkonen and J. T. Kostamovaara. The use of stabilized cmos delay lines for the digitization of short time intervals. *IEEE Journal of Solid-State Circuits*, 28(8):887–894, 1993.

[20] R. B. Staszewski, Chih-Ming Hung, K. Maggio, J. Wallberg, D. Leipold, and P. T. Balsara. All-digital phase-domain tx frequency synthesizer for bluetooth radios in 0.13&mu;m cmos. In *Proc. Digest of Technical Papers Solid-State Circuits Conf. ISSCC. 2004 IEEE Int*, pages 272–527, 2004.

[21] P. Dudek, S. Szczepanski, and J. V. Hatfield. A high-resolution cmos time-to-digital converter utilizing a vernier delay line. *IEEE Journal of Solid-State Circuits*, 35(2):240–247, 2000.

[22] A. K. M. K. Mollah, R. Rosales, S. Tabatabaei, J. Cicalo, and A. Ivanov. Design of a tunable differential ring oscillator with short start-up and switching transients. *IEEE Transactions on Circuits and Systems—Part I: Regular Papers*, 54(12):2669–2682, 2007.

[23] Jianjun Yu, F. F. Dai, and R. C. Jaeger. A 12-bit vernier ring time-to-digital converter in 0.13 um cmos technology. *IEEE Journal of Solid-State Circuits*, 45(4):830–842, 2010.

[24] Poki Chen, Chun-Chi Chen, Jia-Chi Zheng, and You-Sheng Shen. A pvt insensitive vernier-based time-to-digital converter with extended input range and high accuracy. *IEEE Transactions on Nuclear Science*, 54(2):294–302, 2007.

[25] R. Szplet and K. Klepacki. An fpga-integrated time-to-digital converter based on two-stage pulse shrinking. *IEEE Transactions on Instrumentation and Measurement*, 59(6):1663–1670, 2010.

[26] Minjae Lee and A. A. Abidi. A 9 b, 1.25 ps resolution coarse–fine time-to-digital converter in 90 nm cmos that amplifies a time residue. *IEEE Journal of Solid-State Circuits*, 43(4):769–777, 2008.

[27] S. Mandai, T. Iizuka, T. Nakura, M. Ikeda, and K. Asada. Time-to-digital converter based on time difference amplifier with non-linearity calibration. In *Proc. ESSCIRC*, pages 266–269, 2010.

[28] Hayun Chung, H. Ishikuro, and T. Kuroda. A 10-bit 80-ms/s decision-select successive approximation tdc in 65-nm cmos. *IEEE Journal of Solid-State Circuits*, 47(5):1232–1241, 2012.

[29] M. Z. Straayer and M. H. Perrott. A multi-path gated ring oscillator tdc with first-order noise shaping. *IEEE Journal of Solid-State Circuits*, 44(4):1089–1098, 2009.

[30] A. Elshazly, Sachin Rao, B. Young, and P. K. Hanumolu. A 13b 315fsrms 2mw 500ms/s 1mhz bandwidth highly digital time-to-digital converter using switched ring oscillators. In *Proc. IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC)*, pages 464–466, 2012.

[31] Y. Cao, W. De Cock, M. Steyaert, and P. Leroux. 1-1-1 mash delta sigma time-to-digital converters with 6 ps resolution and third-order noise-shaping. *IEEE Journal of Solid-State Circuits*, 47(9):2093–2106, 2012.

[32] 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 Journal of Solid-State Circuits*, 43(7):1666–1676, 2008.

[33] K. Vengattaramane, J. Borremans, M. Steyaert, and J. Craninckx. A gated ring oscillator based parallel-tdc system with digital resolution enhancement. In *Solid-State Circuits Conference, 2009. A-SSCC 2009. IEEE Asian*, 2009.

[34] V. Ramakrishnan and P. T. Balsara. A wide-range, high-resolution, compact, cmos time to digital converter. In *Proc. th Int VLSI Design Held jointly with 5th Int. Conf. Embedded Systems and Design. Conf*, 2006.

[35] A. Mantyniemi, T. Rahkonen, and J. Kostamovaara. A cmos time-to-digital converter (tdc) based on a cyclic time domain successive approximation interpolation method. *IEEE Journal of Solid-State Circuits*, 44(11):3067–3078, 2009.

[36] A. S. Yousif and J. W. Haslett. A fine resolution tdc architecture for next generation pet imaging. *IEEE Transactions on Nuclear Science*, 54(5):1574–1582, 2007.

[37] Wayne C. Goeke. An 8 1/2-digit integrating analog-to-digital converter with 16-bit, 100,000-sample-per-second performance. *Hewlett-Packard Journal*, April:8–14, 1989.

[38] Young-Gyu Yoon, Sang-Hyun Park, and SeongHwan Cho. A time-based noise shaping analog-to-digital converter using a gated-ring oscillator. In *Proc. IEEE MTT-S Int Intelligent Radio for Future Personal Terminals (IMWS-IRFPT) Microwave Workshop Series*, pages 1–4, 2011.

[39] Min Park and M. H. Perrott. A single-slope 80ms/s adc using two-step time-to-digital conversion. In *Proc. IEEE Int. Symp. Circuits and Systems ISCAS 2009*, pages 1125–1128, 2009.

[40] V. Dhanasekaran, M. Gambhir, M.M. Elsayed, E. Snchez-Sinencio, J. Silva-Martinez, C. Mishra, L. Chen, and E. Pankratz. A 20mhz bw 68db dr ct sigma-delta adc based on a multi- bit time-domain quantizer and feedback element. In *Proc. IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC)*, 2009.

[41] Y. M. Tousi and E. Afshari. A miniature 2 mw 4 bit 1.2 gs/s delay-line-based adc in 65 nm cmos. *IEEE Journal of Solid-State Circuits*, 46(10):2312–2325, 2011.

[42] Bernhard E. Boser. Ee lecture 247: Analog-digital interface integrated circuits, filters - outcomes, chapter 11: Sampling circuits. Online, 2011. URL `http://www.eecs.berkeley.edu/~boser/courses/247/lectures/11_sampling_circuits.pdf`.

[43] David Ricketts John A. McNeill. *The Designer's Guide to Jitter in Ring Oscillators*. Springer, 2009.

[44] N. Van Helleputte and G. Gielen. A 70 pj/pulse analog front-end in 130 nm cmos for uwb impulse radio receivers. *IEEE Journal of Solid-State Circuits*, 44(7):1862–1871, 2009.

[45] P. Kinget. *Analog Circuit Design: (X)DSL and Other Communication Systems; RF MOST Models; Integrated Filters and Oscillators*, chapter Integrated GHz voltage controlled oscillators, page 353381. Kluwer, boston, ma edition, 1999.

[46] Paul Geraedts, Ed van Tuijl, Eric Klumperink, and Bram Nauta. Towards minimum achievable phase noise of relaxation oscillators. To be published in International Journal of Circuit Theory and Applications, 2012.

[47] A. Hajimiri, S. Limotyrakis, and T. H. Lee. Jitter and phase noise in ring oscillators. *IEEE Journal of Solid-State Circuits*, 34(6):790–804, 1999.

[48] Guansheng Li, Y. M. Tousi, A. Hassibi, and E. Afshari. Delay-line-based analog-to-digital converters. *IEEE Transactions on Circuits and Systems—Part II: Express Briefs*, 56(6):464–468, 2009.

[49] T. Sepke, P. Holloway, C. G. Sodini, and H.-S. Lee. Noise analysis for comparator-based circuits. *IEEE Transactions on Circuits and Systems—Part I: Regular Papers*, 56(3):541–553, 2009.

[50] W. Sansen. Distortion in elementary transistor circuits. *IEEE Transactions on Circuits and Systems—Part II: Analog and Digital Signal Processing*, 46(3):315–325, 1999.

[51] R. H. Walden. Analog-to-digital converter survey and analysis. *IEEE Journal on Selected Areas in Communications*, 17(4):539–550, 1999.

[52] B. Murmann. Adc performance survey 1997-2012. Online, October 2012. URL `http://www.stanford.edu/~murmann/adcsurvey.html`.

[53] Xiang Gao, B. Nauta, and E. A. M. Klumperink. Advantages of shift registers over dlls for flexible low jitter multiphase clock generation. *IEEE Transactions on Circuits and Systems—Part II: Express Briefs*, 55(3):244–248, 2008.

[54] A. A. Abidi. Phase noise and jitter in cmos ring oscillators. *IEEE Journal of Solid-State Circuits*, 41(8):1803–1816, 2006.

[55] Ken Kundert. Simulating switched-capacitor filters with spectrerf. Online, July 2006. URL `www.designers-guide.org`.

# A List of performance metrics for ADCs and TDCs

Since many ADC techniques have their direct counterpart in TDCs, many well-known performance metrics are applicable on both ADC and TDC level.

Furthermore, TDCs can be tested with sinusoidal input signals just like ADCs, for instance by modulating a pulse width by a sinewave. This makes the performance metrics not only of theoretical, but also of practical importance for TDCs.

The following performance metrics can be defined for reference purposes:

- *Dynamic range (DR):* Ratio between the power of a uniform distribution of input values and the power of the converter noise floor. For an ideal $N$-bit quantizer, $DR = 6.02 \cdot N$. For a thermal noise limited quantizer, the dynamic range is lower. Harmonic distortion products are excluded from the calculation.

- *Signal-to-Noise ratio (SNR):* Ratio between the power of a full-scale sinewave input and the power of the converter noise floor. Typically 1.761 dB higher than the DR.

- *Total harmonic distortion (THD):* Ratio between the power of a full-scale sinewave input and the combined power of all harmonics.

- *Signal-to-Noise-and-Distortion ratio (SINAD or SNDR):* Ratio between the power of a full-scale sinewave input signal and the power of all other products, including noise and distortion.

- *Effective number of bits (ENOB):* The number of bits in an ideal quantizer that would yield the same SINAD as the measured one. Calculated using $\text{ENOB} = \frac{\text{SINAD} - 1.761\text{dB}}{6.02}$.

- *Figure-of-Merit (FoM):* The average amount of energy required to calculate one level in time or amplitude. This work will use the Walden FoM [51], defined as $\text{FoM} = \frac{P}{2^{\text{ENOB}} \cdot f_s}$, where $P$ is the power consumption of the converter and $f_s$ is the sampling frequency.

- *Integral nonlinearity (INL):* The deviation of the converter transfer function (time to output code) from a linear fit, expressed in LSBs.

- *Differential nonlinearity (DNL):* The derivative of the INL, expressed in LSBs.

- *Gain error:* The ratio between the average converter transfer function gain and the desired gain.

- *Offset:* The offset of the converter transfer function from the desired transfer function.

- *Non-monotonicity:* An imperfection in the converter that leads to an increasing signal generating a decreasing code.

- *Missing code:* An imperfection in the converter that leads to an output code that cannot be generated by any input value.

- *Power consumption:* The average power consumption of the converter from the supply.

- *Area:* The silicon area occupied by the converter.

- *Resolution:* The range of input values represented by one output code, although a 'higher resolution' implies a smaller range represented by one code.

Besides performance metrics that can be applied to both ADCs and TDCs, there is one TDC-specific performance measure that is worth mentioning:

- *Single-shot precision (SSP):* The variance of the output code when a static time input is applied, expressed in LSBs.

## B   Noise and matching of inverter-based delay elements

For current-mode logic, a jitter-power FoM has been defined in [53]. This FoM demonstrates that for a CML delay stage, when all topological choices are made, both thermal and mismatch jitter are proportional to the power and the realized delay.

For a full-swing transition through a CMOS inverter, the thermal jitter in the propagation delay $t_d$ is approximately given by [54]:

$$\sigma_{td}^2 = \frac{4kT\gamma\, t_d}{I(V_{dd} - V_{th})} \tag{13}$$

where $I$ is the saturated current of the active transistor, and $\gamma$ is its noise factor. Given a load capacitance $C$ and neglecting the crowbar current, the power consumed to realize $t_d$ is

$$P = \frac{CV_{dd}^2}{2t_d} \tag{14}$$

and the delay until the output crosses $V_{dd}/2$ can be written in terms of $C$ and $I$:

$$t_d = \frac{CV_{dd}}{2I} \tag{15}$$

combining 13, 14 and 15 and normalizing the jitter by $P/(1\text{mW})$ to compensate for admittance scaling, yields:

$$\sigma_{td,normalized}^2 = \frac{4kT\gamma t_d V_{dd}}{(V_{dd} - V_{th})(1\text{mW})} = t_d \cdot c \cdot \frac{V_{dd}}{1\text{mW}} \tag{16}$$

which shows that for CMOS inverters, the thermal jitter per power is also proportional to the realized delay.

# C  Successive approximation TDC concept

An early idea in this assignment was to create an unrolled successive-approximation (SAR) TDC. The mechanism for a 5-bit TDC using this concept is shown in figure 54. In contrast to the main part of this thesis, the image shows a 'bipolar' input: the 'input' edge can arrive at in a time span centered around the reference edge. An arbiter or phase detector detects which edge arrives first and sets a multiplexer to send the earlier edge into a delay of 8 units. After that, the same process is repeated for 4, 2, 1 units. Buffer units are required to delay the edges while the phase detectors decide. After the decision of bit 4, the edges are aligned with an accuracy of below one gate delay. These aligned edges could be sent to for instance a VDL TDC to resolve even more bits.

The five bits can also be used to set a second path such that the reference edge will experience the delays that it skipped until now. The result is a reference edge that should always experience the same net delay. This output reference edge could be used in for instance a locked loop to tune the delays, or in some sort of calibration procedure.
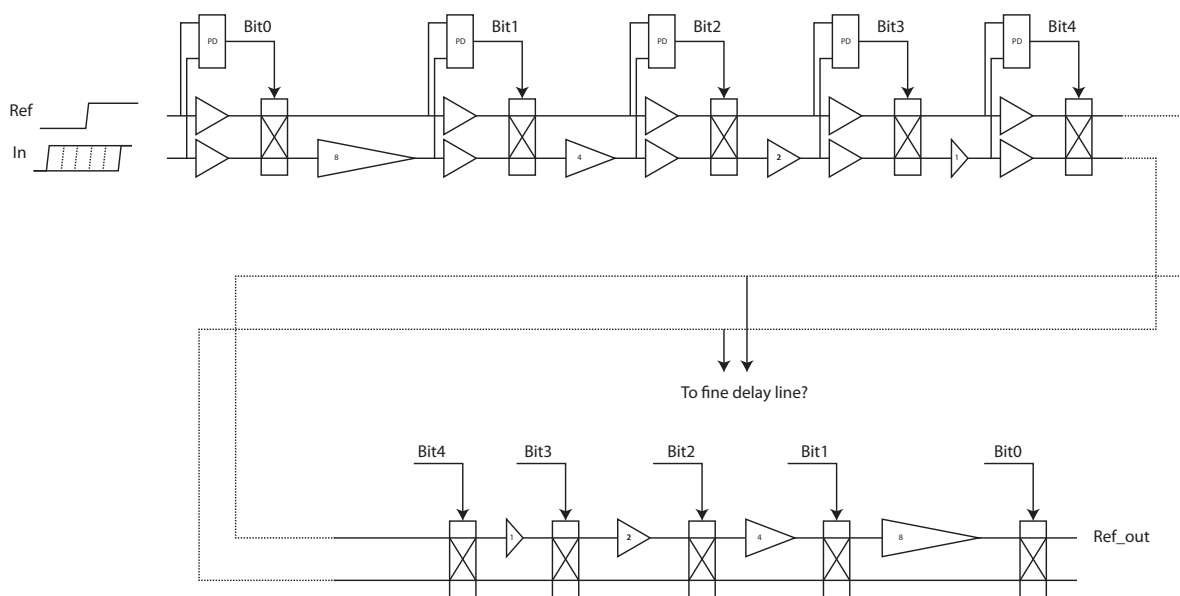


**Fig. 54:** Illustration of an early proposal for a SAR TDC.

The idea stemmed from the following observation: When increasing the dynamic range (number of stages) of a flash TDC, the deterministic jitter due to mismatch starts impairing the linearity long before the effects of thermal jitter become visible. However, there is no way to 'worsen' the thermal jitter performance of the individual stages without slowing down the entire TDC. In a SAR TDC, the delays are lumped into 1, 2, 4, 8, ... 256, 512 ... and these larger delays may offer more possibilities to trade power for noise performance.

This is reinforced by the observations from appendix B. It is shown that several types of delay elements have a fixed jitter-power figure of merit. In other words, to implement a certain amount of delay with a certain amount of jitter, a fixed amount of power is required. So if there is jitter to spare, power can be gained. Large delays such as 256 unit delays can now be implemented for instance as 128 current-starved delays drawing half the power.

There is a number of practical reasons why this SAR TDC concept was not carried out further. The main reason is the issue of metastability: the last arbiters have to resolve time differences in the order of a few LSBs every clock

cycle. Therefore their metastability behavior is crucial: if the arbiters cannot decide within the buffer delay time, the multiplexer is not set in time, and the sample is lost for the next stages. A thorough metastability study and arbiter design did not fit within the time span of this thesis.

Furthermore, there are indications that mismatch-jitter will scale with the same quantities as thermal jitter, so effectively nothing may be gained by current-starving the long delays [53], but this was not verified for specific delay implementations, such as current-starved inverters, within this thesis.

# D Spectre RF time domain noise analysis

Spectre RF time domain noise analysis [55] uses the periodic steady state (PSS) of a periodically operating circuit to determine the voltage uncertainty between any two nodes at any point during the fundamental period. Naturally, it requires the periodic steady state (PSS) of the circuit to be known.

The analysis will be executed for each time point of interest during the period, and for each noise frequency at that time point. It starts from an instant somewhere in the periodic steady state and cycles through the PSS to find the accumulated noise at that frequency. It accounts for folding (aliasing) of noise in sampled circuits into the band from 0 to half the fundamental frequency.

Knowing the bandwidth of the relevant noise sources is the key to correct simulation results. An example will follow, but suppose the system has a fundamental frequency (e.g. the clock frequency) of $f_{\text{fund}}$. Suppose the noise source of interest band-limited to a frequency $f_{\text{noise}} = N \cdot f_{\text{fund}}$. To find reasonable initial results, the following procedure may be followed in setting up the analysis.

First of all, the PSS needs include sufficient harmonics, to say anything about what happens at higher frequencies (that will fold back into the signal band of a sampled system). Therefore, the number of harmonics must be set to above N. Secondly, the PSS needs to be sufficiently accurate to include these high frequencies. Therefore, in the options of the PSS settings, the parameter 'maxac' must be set to above $f_{\text{noise}}$.

Next, a 'pnoise' analysis must be enabled, with the noise ranging from the lowest value of interest (e.g. 1 Hz) to half the fundamental frequency $f_{\text{fund}}$, in for instance 10 steps per decade. Higher frequencies do not need to be included, even if $f_{\text{noise}}$ is much higher, since the pnoise analysis will automatically account for noise folding. Also, the parameter 'maxsideband' must be set to above N, but slightly smaller than the number of harmonics included in the PSS simulation. Finally, the number of time points must be specified at which the integrated noise should be calculated (e.g. 20 points during a full period). Specific instants can also be listed, such as the time just before or after a switch opens or closes in a sampled system.

## D.1 Applying the analysis to the VTC

The aforementioned procedure was applied to the VTC in section 5.2.2. This section lists the specifics of that analysis.

The noise sources with the highest bandwidth in this circuit are the sampling switches. Their resistance at a 1V input is approximately 1.35 kΩ, which, together with the sampling capacitance of 556 fF implies a noise bandwidth of 212 MHz. Another relevant noise source is the current source, however, it has a lower bandwith, since its high output impedance forms a low-pass filter with the sampling capacitance. Also, the bias line of the current source is decoupled to the supply rail to limit the noise bandwidth.

The test circuit for noise testing purposes is actuated by a clock signal of 5 MHz; this is the PSS fundamental frequency. Since the noise extends to 212 MHz or 42 times the fundamental, at least 42 harmonics of the fundamental need to be included in the PSS simulation.

To generate the relatively detailed plots of figure 50, loose simulation parameters were chosen: 50 harmonics, 48 sidebands and 'maxac' = 240 MHz.

Since the plots showed a deviation of approximately 14% below the expected values, more conservative simulation settings were used for subsequent simulations that required only two time points. For these simulations, 100 harmonics, 'maxac' = 500 MHz and 95 sidebands were used. Deviations were now only a few percents below the expected values.

# E MATLAB code

## E.1 Process flip-flop Monte-Carlo simulations

```matlab
% Calculates offset of flipflops based on montecarlo simulations for
% different input voltages

% Probability that flipflop samples the voltage as positive
Pv = [0.01 0.06 0.16 0.34 0.57 0.75 0.91 0.98 1];
% corresponding input voltages
V = [0.83:0.01:0.91];
Ntrials = 100;

% Probability that flipflop samples edge as positive
Pt = [0 0.01 0.03 0.07 0.13 0.18 0.28 0.35 ...
    0.45 0.59 0.66 0.82 0.9 0.95 0.98 0.99 1];
% Corresponding input times
T = [337:1:353].*1e-12;

% Choose the input to be used in the remainder of the script
Xin = T;
Yin = Pt;

% Create matrix with column of successes and column of trials
Y = [Yin'*Ntrials repmat(Ntrials,size(Yin'))];

% Fit binomial distribution, store result
b = glmfit(Xin,Y,'binomial','link','probit');

% Pick some points at which to evaluate the fitted curve
refx = linspace(min(Xin),max(Xin),50);

yfit = glmval(b,refx,'probit');
figure
hold on

% Plot the sampled distribution
plot(Xin,Yin,'ob');
% Plot the fitted distribution
th = plot(refx,yfit,'--r');
% Plot the corresponding normal distribution
mu = -b(1)/b(2)
sigma = 1/b(2)

%line(refx,normcdf(refx,mu,sigma),'color','g');

box on
hold off

legend('Mismatch simulations','binomial distribution fit','Location','East');
xlabel('Clock delay (sec)');
ylabel('Probability of positive output');
title('INL of delay line vs delay ring');
```

## E.2 Calculate INL profiles of a ring and linear TDC

```matlab
delay = 90.7e-12;      % Nominal delay of elements
sigma_rel = 0.027;    % Relative stddev of delay elements
sigma_thermal_rel = 0.00077; % Random deviation of elements (thermal noise)
ringstages = 18;     % Number of stages in the ring implementation
bits = 18;           % Number of bits to be calculated
numplots = 1;        % Number of randomizations to plot

% Calculate absolute sigma
sigma = delay*sigma_rel;
```

```matlab
10  sigma_thermal = delay*sigma_thermal_rel;
11
12  % Calculate the number of "laps" the signal can go round in the ring
13  laps = ceil(2^bits/ringstages);
14
15  hold off;
16  plot(1:1:laps*ringstages,delay*ones(1,laps*ringstages),'k--', ...
17      1:1:laps*ringstages,-delay*ones(1,laps*ringstages),'k--');
18  ylim([-2*delay 2*delay]);
19  xlim([1 laps*ringstages]);
20  hold on;
21
22  for I = 1:numplots
23      % Straight line is accumulation of independent delays
24      Nomline = (1:1:laps*ringstages).*delay;
25      Sigmaline = cumsum(sigma*randn(1,laps*ringstages)+sigma_thermal*randn(1,laps*ringstages));
26      Qline = Nomline + Sigmaline;
27
28      % Ring is accumulation of repeated delays
29      Nomring = (1:1:laps*ringstages).*delay;
30      Sigmaring = cumsum(repmat(sigma*randn(1,ringstages),1,laps) ...
31          + sigma_thermal*randn(1,laps*ringstages));
32      Qring = Nomring + Sigmaring;
33
34      % Calculate fit for slope and offset correction
35      A = polyfit(1:1:laps*ringstages,Sigmaring,1);
36      B = polyfit(1:1:laps*ringstages,Sigmaline,1);
37
38      plot(1:1:laps*ringstages,Sigmaring-A(1)*(1:1:laps*ringstages)-A(2),'b-', ...
39          1:1:laps*ringstages,Sigmaline-B(1)*(1:1:laps*ringstages)-B(2),'r-');
40  end
41
42  xlabel('Code');
43  ylabel('INL (sec)');
44  title('INL of delay line vs delay ring');
```