POLITECNICO
MILANO 1863

Diparimento di Elettronica, Informazione e Bioingegneria
Doctoral Program In Information Technology

Sub-μA Polysilicon MEMS Real-Time Clock with Deterministic Jitter Compensation

Doctoral Dissertation of:
Giorgio Mussi

Supervisor:
Prof. Giacomo Langfelder

Tutor:
Prof. Carlo Fiorini

The Chair of the Doctoral Program:
Prof. Barbara Pernici

2019/20 – Cycle XXXII

Abstract

Real-Time Clocks have been object of intense studies for about a century, with innovations both on the resonator side and in the related electronics. Recent advancements have seen the introduction of MEMS devices as the fundamental resonant element, fabricated with tailored, dedicated and expensive processes. This Thesis proposes an exploration of the advantages – and limits – of relying on MEMS resonators built with a standard, large-scale and cheap epitaxial polysilicon process, regardless of their intrinsic 20-fold poorer thermal stability, that is ±2000 ppm versus ±100 ppm over the temperature range. The ambitious proposal is based on the hypothesis – verified over the course of the project – that polysilicon devices are characterized by a thermal frequency instability that is more repeatable than their custom-made counterparts, thereby achieving a great reduction of long calibration procedures in an industrial, large-scale application. The implications of worse thermal properties at system-level are then investigated in detail and the overall feasibility of the integrated system, based on a ΣΔ-modulated digital divider, is assessed, both in theory and on silicon, under the constraint of a power budget of 1 µA. Additionally, the spurious phase noise due to the ΣΔ operation is suppressed with an alternative, low-power approach that, to the author's knowledge, had never been implemented at hundreds of nW. The designed integrated circuit is tested to prove the implemented concepts.

Contents

List of Publications

This Thesis is based on the following peer-reviewed journal paper:

- **G. Mussi**, M. Bestetti, V. Zega, A. Frangi, G. Gattere and G. Langfelder, "An outlook on potentialities and limits in using epitaxial polysilicon for MEMS real-time clocks", *Transactions on Industrial Electronics*, 2019 [1]

and has been presented at the following international conferences:

- **G. Mussi**, M. Bestetti, V. Zega, A. Frangi, G. Gattere and G. Langfelder, "Resonators for real-time clocks based on epitaxial polysilicon process: A feasibility study on system-level compensation of temperature drifts," *2018 IEEE Micro Electro Mechanical Systems (MEMS)* [2]
- **G. Mussi**, P. Frigerio, G. Langfelder and G. Gattere, "A Low-Power Deterministic Approach to Jitter Suppression in MEMS-Based Real-Time Clocks," *2019 Joint Conference of the IEEE International Frequency Control Symposium and European Frequency and Time Forum (EFTF/IFCS)* [3]

Moreover, a patent was filed regarding a sub-block of the implemented integrated circuit:

- **G. Mussi**, G. Langfelder, G. Gattere and C. Valzasina, "Generatore di orologio in tempo reale a potenza ultra bassa utilizzante un risonatore microelettromeccanico e metodo di compensazione di jitter in tale generatore di orologio", Patent Application IT2018000010577, 2019 [4]

CHAPTER 1

Real-Time Clocks: What Are They?

Real-Time Clocks (RTC) are key components in many applications where a notion of time must be shared with accuracy between different agents. This Chapter introduces the main requirements of a RTC and explains them with a typical application case. The mature quartz resonators and their rampant MEMS counterparts are compared, identifying the pros and cons of each. The stability of the output frequency versus temperature is identified as one of the main system requirements, therefore the main strategies of compensation of the temperature drift are reviewed.

1.1 Introduction

Nowadays timing landscape presents challenges that are mainly driven by two distinct fields of applications: 5G and the Internet of Things (IoT).

On one hand, 5G is pushing consumer applications towards frequencies that were once uncharted territory. In this context, the focus is posed on obtaining good noise performance at very high frequency and at a power level sustainable by, e.g., common smartphone batteries.

On the other hand, key enabling features for the IoT are very small power consumption, at the µW level or even below, and package volume/footprint.

A widespread use of small, autonomous and interconnected nodes includes cases where such nodes are placed in remote areas where they cannot be replaced or where their batteries cannot be replaced. Besides, a smaller required PCB area may enable new applications where the size of the node is critical.

This Thesis tackles the timing problem in the latter context. Specifically, the goal is to assess the feasibility of new-generation Real-Time Clocks (RTC). These clocks are the typical choice for time-keeping applications, where a low-frequency oscillator is needed to keep track of the passing of time, e.g. to synchronize operations between different systems, at low power consumption. Historically, for this purpose oscillators based on a quartz resonant element have been used, running at a frequency of 32.768 kHz. This choice was made as a compromise between size, pushing for higher frequencies, and power consumption, demanding lower frequencies. Moreover, a frequency of 1 Hz can be obtained by halving this frequency 15 times: a very convenient feature for wristwatches, an early application for this kind of devices. As a matter of fact, this frequency is the *de-facto* industry standard that must be adopted by any manufacturer.

To understand more in detail the problems and challenges faced in this field, the following section describes a common use-case for RTCs of the last and next few years.

1.2 Application Case and Specs

Virtually every computing system that interacts with the physical world needs a notion of time. In fact, the application requirements may dictate periodic actions, such as the read-out of a sensor, the activation of a radio to wait for a packet of data or more in general a wake-up from sleep mode to perform some operation. Another requirement may consist in the so-called *time-stamping*, where the occurrence of an event, such as the reception or transmission of a data packet, an interrupt request. . . , has to be associated with the time instant at which it occurred.

An interesting use case is that of *Wireless Sensor Networks* (WSN). A WSN is an ensemble of connected sensor nodes able to sense physical quantities, process the gathered data and communicate them to other nodes. Thus, they are typically equipped with a sensor, a (low-power) Micro-Controller Unit (MCU) and an RF transceiver. The whole idea is based on the hypothesis that having many widespread nodes able to cooperate is more convenient than having a single expensive sensor. These sensors can be deployed in vast amounts and in regions that are hard to reach, e.g. in

military or biomedical applications [5]. Therefore, two main requirements arise in these applications:

- Low cost: so that the cost of these many cheap sensor nodes does not overcome that of one single and more powerful sensor.
- Low power consumption: so that one can run the node for as much time as needed from a single, maybe irreplaceable, battery.

Both these constraints are addressed within this Thesis, as described in the following.

1.2.1 RTCs and Power Consumption in WSNs

The level of low-power operation that the nodes of such networks aim to achieve are usually incompatible with power hungry blocks such as the MCU and the radio. Therefore, the operations carried out by the node are heavily duty-cycled. In this way, for most of the time the radio is off and the MCU can enter the sleep mode. A very simple model to estimate the average current consumption $\langle I \rangle$ of this system is the following:

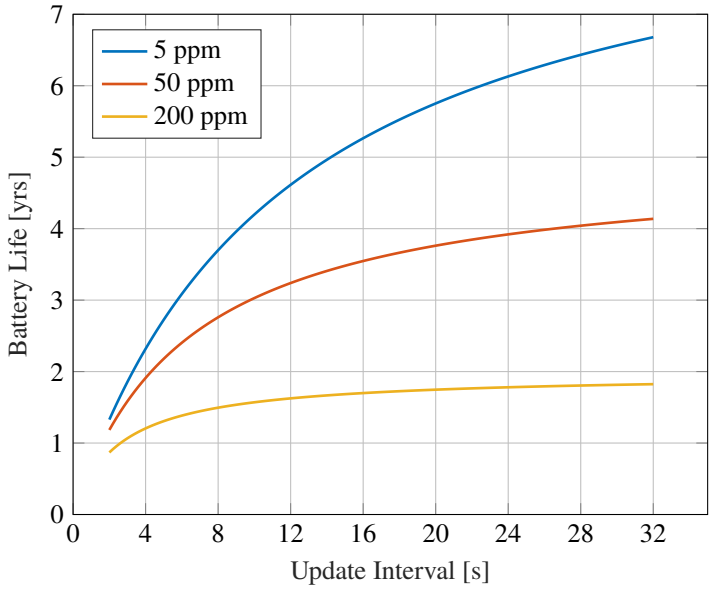
$$\langle I \rangle = I_{off} + I_{on} \cdot \frac{t_{on}}{T} + I_{rx} \cdot \frac{t_{rx}}{T} \tag{1.1}$$

where I_{off} is the consumption during the off phase, I_{on} and t_{on} are the consumption and the duration of the on phase, I_{rx} and t_{rx} are the consumption and on-time of the radio and T is the period of the duty cycle.

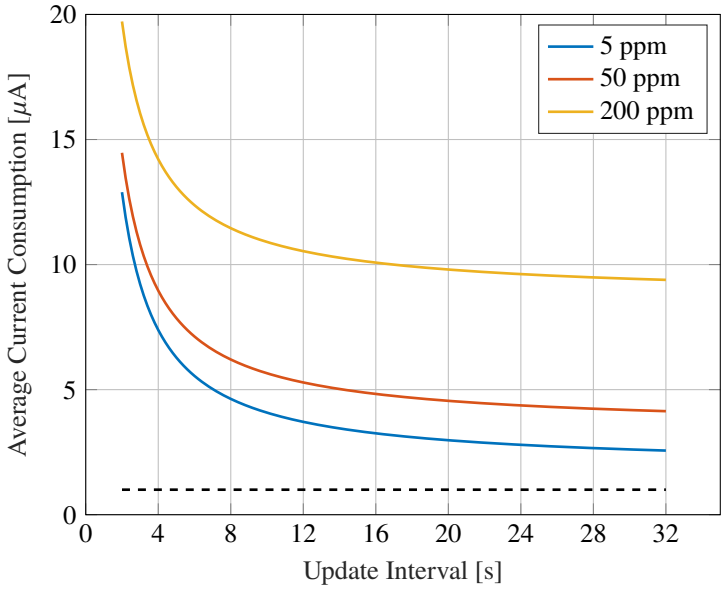
Let us consider all the unknowns of (1.1) determined – by either the required sample rate or the component cost – apart from t_{rx}. This time interval is required to be as long as the data burst duration t_{data} plus some additional time to accommodate for the timing uncertainty t_ϵ. The uncertainty stems from the fact that the time measured by two different nodes – by their own RTCs, indeed – may differ by up to few hundreds of ppm. This is caused primarily by a difference in their temperature, but also from production imperfections, aging and humidity. Therefore, given a certain maximum relative frequency variation $\Delta f / f$, we can estimate t_ϵ as:

$$t_\epsilon = 2 \cdot \frac{\Delta f}{f} \cdot T \tag{1.2}$$

In this way, we can express the power consumption of the node as a



(a) *Battery life.*



(b) *Current consumption. The dotted black trace is* $I_{off} = 1\,\mu A$*, i.e. the asymptotic consumption if the update interval were infinite.*

Figure 1.1: *Battery life and current consumption of a WSN node powered by a* 150 mAh *low-footprint coin cell for three different clock accuracies as described by* (1.4)*. Supposing a* 20 ppm *accuracy for the master, a* 5 ppm *RTC can double the battery life of the node compared to a* 50 ppm *clock. The values used for the plots are taken from [6] and are:* $I_{on}t_{on} = 18.7\,\mu C$*,* $I_{off} = 1\,\mu A$*,* $t_{data} = 115\,\mu s$*,* $I_{rx} = 17.5\,mA$*.*

function of their RTC accuracy $\frac{\Delta f}{f}$ by combining (1.1) and (1.2) as follows:

$$\langle I \rangle = I_{off} + I_{on} \cdot \frac{T_{on}}{T} + I_{rx} \cdot \frac{t_{data} + t_{\epsilon}}{T} = \tag{1.3}$$

$$= I_{off} + I_{on} \cdot \frac{T_{on}}{T} + I_{rx} \cdot \left(\frac{t_{data}}{T} + 2 \cdot \frac{\Delta f}{f} \right) \tag{1.4}$$

Specifically, in this discussion the last term $2I_{rx}\frac{\Delta f}{f}$ is the most relevant. If the wake-up period T is long enough, then the RTC accuracy may play a very significant role and increase the power consumption to unacceptable levels. Figure 1.1a highlights this phenomenon by plotting the battery life of a realistic node for wake-up periods ranging from 2 s to 32 s and different clock accuracies. What emerges from the plots is that a better RTC frequency accuracy can lead to a terrific power saving, to a scale that could possibly enable new applications.

Moreover, when the clock accuracy is good enough and the update interval is sufficiently large, the current consumption is dominated by the term I_{off}, as highlighted by the plot in Figure 1.1b. Even a small RTC current consumption of 1 µA can dominate the total power consumption in these conditions. Hence, the strong requirement on the consumption of these devices.

1.2.2 RTC Jitter in WSNs

The previous discussion highlighted the role of the RTC as an always-on time-keeping device. This role and the time scales apparently imply no particular jitter requirement, since the measurement error on a long-term time interval is typically dominated by the frequency inaccuracy, rather than the accumulation of the short-term zero-mean jitter.

Indeed, the entity of the clock jitter is more important when the RTC is used for other kinds of operations, such as the aforementioned time-stamping. When one wants to time-stamp an event with a simple digital counter, the resolution is given by the clock period, which is equal to 30.5 µs for typical RTCs. If a better resolution is desired, one can use the high-frequency clock – 40 MHz is a typical value – that is always available as it is required by the radio of the node. The disadvantage of this simple solution is that now a power-hungry peripheral is always turned on, with ill-fated consequences on the battery life of the node.

Smarter strategies have been proposed, which are based on the combined use of the high-frequency clock and the low-frequency RTC to obtain the best of both: a good resolution at low power consumption [7], [8]. The

Parameter	Value	Units
$\Delta f/f$	±10	ppm
I_{tot}	1	µA
σ_j	35	ns_{rms}
f_{out}	32.768	kHz

Table 1.1: *Summary of the performance for the RTC system.*

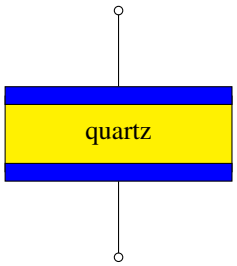


Figure 1.2: *Simple model of a quartz resonator.*

idea consists in combining the coarse count of a digital counter clocked by the RTC with the count of a second digital counter that is clocked by the high-frequency clock. The latter is operated with a narrow duty-cycle as it is turned on only when the event is expected to happen. Therefore, its impact on the power consumption can be made negligible.

The achieved time-stamping resolution is therefore equal to the high-frequency clock period. Nevertheless, one cannot ignore the jitter of the slow RTC, which may be small when compared to the RTC period (30.5 µs) but significant when compared to the 40 MHz period (i.e. 25 ns). If this jitter is much larger, then the resolution is not optimal, limited by the RTC jitter rather than the quantization given by the high-frequency clock.

1.2.3 Target Requirements

In view of the preceding discussion, the key requirements for the development of this Thesis reported in Table 1.1 were agreed with the industrial partner of this project. The frequency accuracy $\Delta f/f$ and the total consumption I_{tot} are two universal performance indicator for any RTC application, whereas the output jitter σ_j is a nice-to-have option that enable a RTC to cover more applications.

Figure 1.3: *A tuning fork quartz crystal. The element is, indeed, a tuning fork made out of a quartz blank with deposition of a metallic material that forms the actuation and sensing electrodes.*

1.3 Quartz vs MEMS

Historically, the RTC research and market have been dominated by Quartz-based oscillators for about a century. Indeed, these components were relatively inexpensive and showed extremely good properties.

Quartz resonators are electro-mechanical elements that are built out of a bank of crystalline quartz. This material is very common on Earth and shows piezo-electric properties, i.e. it gets strained when a voltage is applied to it, and it gets electrically polarized when it is strained. Therefore, one can build a simple resonator as illustrated in Figure 1.2. Such a device is formed by a piece of quartz that forms the dielectric of a parallel-plate capacitor, whose electrodes are placed on the top and the bottom of the device.

This device is able to store energy in its electrical domain: a fraction of this energy turns into mechanical energy, thanks to the relationship between electrical voltage and mechanical strain. The maximum stored mechanical energy occurs at the resonance frequency of the device. This resonator typically shows excellent properties such as high quality factors and low equivalent series resistances. These parameters vary a lot according to the crystal cut orientation and the operating frequency.

The most common crystal for RTC 32 kHz applications is the X-cut tuning fork design, that is depicted in Figure 1.3. These devices are characterized typical quality factors in the order of 10 000 and series resistances

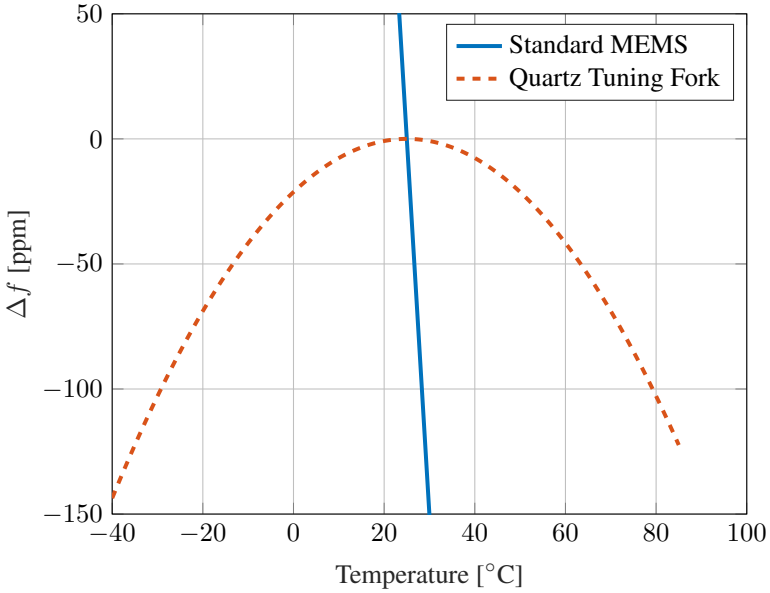


Figure 1.4: *Resonant frequency drift comparison between MEMS and Quartz devices.*

around 90 kΩ. They show a temperature dependence of the resonant frequency $f(T)$ that is flat at room temperature with a second order Temperature Coefficient of frequency $TC\,f_2$ of $-0.034\,\text{ppm/K}^2$. As shown in Figure 1.4, this translates in a drift of the RTC frequency of 140 ppm across the −40 °C to 85 °C temperature range.

Nonetheless, in the last decade the quartz industry has seen an increasing competition from MEMS devices. Micro-Electro-Mechanical Systems leverage the same concept at the basis of the quartz resonator success: by means of a transduction mechanism between the mechanical and electrical domains it is possible to achieve resonators with precise frequencies and large quality factors, to a degree unreachable by electronic resonators.

On one hand, MEMS devices suffer from a few problems with respect to quartz devices. Firstly, they exhibit an approximately linear $f(T)$ with a very large first-order Temperature Coefficient of frequency $TC\,f_1 \approx -30\,\text{ppm/K}$, compared in Figure 1.4 to that of quartz devices. This directly translates in a frequency inaccuracy of about ±1800 ppm over the same temperature range, hence some compensation mechanism is essential. Secondly, MEMS devices show typically a worse power handling than quartz resonators. As will be discussed in the Thesis, a few constraints on the integrated oscillator arise because of this limitation.

On the other hand, MEMS devices have a few advantages over their

quartz counterparts. Firstly, the fabrication process of MEMS devices is quite similar, even though not fully compatible, to that of CMOS integrated circuits. The CMOS-friendliness can be leveraged to make use of sophisticated ad-hoc RTC integrated circuits to perform background compensation in a dedicated System-in-Package (SiP), with no need of external components. Quartz oscillators, instead, usually require external passive components to form the low-power oscillator, namely two capacitors and, possibly, a resistor. Secondly, MEMS technology is already very established, with mass-production devices already on the market since the 90s. Several companies can potentially start a competition with quartz manufacturers re-using existing fabs, hence with relatively low initial economical overhead. Thirdly, MEMS devices are less susceptible to vibrations, which can break the device over years of operation [9].

Up to the last few years, the pros MEMS technology had to offer were not sufficient to begin a transition on a large scale. But lately, miniaturized low-power applications have stressed more the requirements in terms of size and power consumption. This fact combined with some solutions to the very large native frequency drift of MEMS resonators, arguably the biggest challenge, led MEMS-based RTCs to a significant commercial success [10].

1.4 Thermal Drift and Compensation Strategies

Standard[1] MEMS resonators do not guarantee the degree of thermal stability required for RTC applications. This is very clear by comparing the magnitudes of the frequency drifts shown Figure 1.4. Therefore, some kind of compensation must be enacted.

The main approaches to the thermal stabilization of a resonator frequency are (i) process-level compensation, (ii) system-level compensation and (iii) ovenization of the resonator.

The third option consists in heating the resonator to a stable temperature, larger than that of the surrounding environment. This technique will not be mentioned further as it is typically used in applications where the focus is on performance over consumption [11], unlike in the field of RTCs. Therefore, the first two options will be analyzed in the following.

1.4.1 Process-Level Compensation

The cause of the $-30\,\mathrm{ppm/K}$ coefficient lies in the temperature dependence of the Young modulus of undoped silicon E, approximately equal

[1] In this Thesis, a MEMS process referred to as "standard" means a polysilicon process characterized by a relatively large Temperature Coefficient of Frequency of $\approx -30\,\mathrm{ppm/K}$.

to $-60\,\mathrm{ppm/K}$. Due to the proportionality between Young modulus and equivalent stiffness k:

$$f_0 = \frac{1}{2\pi}\sqrt{\frac{k}{m}} \propto \sqrt{E} \Rightarrow \frac{\Delta f}{f} = \frac{1}{2}\frac{\Delta E}{E} = -30\,\mathrm{ppm/K}$$

A quite intuitive way to compensate this temperature dependence consists in using a composite material formed by silicon, the usual MEMS structural layer, and another material characterized by an opposite TCf_1. An example of such a material is silicon dioxide (SiO_2), which is also compatible with typical MEMS processes. This technique has been exploited in a few works, even outperforming the quartz stability with an accuracy of 100 ppm across the temperature range [12].

Another possibility is to exploit the dependence of the elastic constant of crystalline silicon on its doping level, a phenomenon that had already been highlighted by Hall in the 60s [13]. In this case, the main degrees of freedom allowed to the designer are the choice of the dopant element, the doping level and the mechanical design orientation with respect to the crystallographic axes. It has been proven that with a near-degenerate doping of the structural layer the TCf_1 can be greatly reduced. An overall frequency drift of few hundreds of ppm on the temperature range for doping levels in the vicinity of $1 \times 10^{20}\,\mathrm{cm}^{-3}$ is achievable [14].

1.4.2 Electronic Compensation

Even though these techniques are effective in reducing the frequency drift to levels comparable to that of quartz devices, an additional level of compensation is required in order to outperform traditional RTCs. This extra layer can be implemented in an integrated circuit that, in a final application, comes embedded in the same package together with the MEMS resonator. Unless each part is trimmed, this layer is necessary even in the case of natively-compensated resonators, since the nominal frequency of each sample is characterized by its production spread.

The most popular technique exploits a mixed-signal electronic system that makes use of a pulse-swallow frequency divider (also called fractional frequency divider) and a temperature sensor at the same temperature of the resonator [15]–[17]. The block scheme of the most basic version of the architecture is sketched in Figure 1.5.

In this case, an oscillator based on a MEMS resonator is used as a frequency reference. Its frequency, indicated as f_0, is higher than the desired 32.768 kHz output and may drift as either a standard MEMS resonator

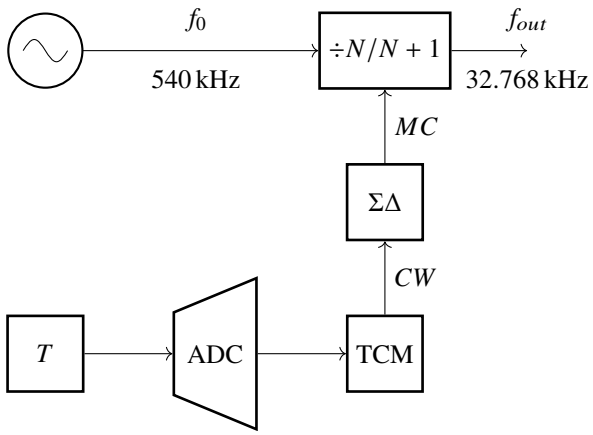


Figure 1.5: *Basic electronic architecture used to compensate the frequency drift according to the measured temperature. The oscillator block may be based on a natively-compensated non-standard MEMS resonator, thereby combining the two macro strategies described. The temperature sensor is supposed at the same temperature of the resonator (the whole integrated circuit is usually in contact with the MEMS die).*

(±1800 ppm), or as a natively-compensated one (e.g. ±300 ppm). The correct output frequency f_{out} is then achieved with the use of the multi-modulus divider that is able to divide the input frequency f_0 by either N or $N + 1$ according to its Modulus Control input (MC). For example, one can choose to nominally divide once by N and subsequently once by $N + 1$ in a perpetual cycle, so that he is dividing by $N + 0.5$ *on average*. When a change in temperature is sensed, the proportion between N and $N + 1$ divisions is changed so that – ideally – no variation at the output can be noticed. Therefore, it is necessary that the temperature of the resonator is sensed – *somehow* – and digitized by the T and ADC blocks. The digital word thus produced is then processed by a Temperature-Compensating Machine (TCM) that encodes in a multi-bit Command Word (CW) the fractional part of the division modulus. The CW is then transformed into the single-bit MC signal that drives the fractional divider by a Digital ΔΣ modulator. The ΔΣ modulation is made possible thanks to the availability of a clock much faster than the expected temperature variations, so that the temperature can be sensed only a few times per second and the average value of the MC can correctly encode the CW value. With this technique, a calibration of the production spread of each part can be easily introduced: the resonance frequency of the sample is measured at room temperature and a fixed term is added to CW, so that the output frequency is centered at 32.768 kHz.

Temperature Sensors

In the literature, two main approaches to temperature sensing are reported.

One, perhaps the most obvious, consists in using a traditional integrated temperature sensor employing a bipolar transistor to produce a current proportional to absolute temperature (PTAT). The sensor is then followed by an ADC. This technique was used in [17] where the authors used a $\Sigma\Delta$ ADC and dynamic element matching techniques, reporting a resolution of 25 mK and a Figure of Merit of 24 pJ $\times$ °C^2, which translates into a current consumption of 150 nA at a 3 Sa/s sampling rate.

Another approach consists, instead, in the use of two different oscillators whose oscillating frequencies have different – or even opposite – TCf. In [15], [16] an additional relaxation oscillator was integrated on the IC, and it was used as the reference of a counter that measured the frequency of the main MEMS oscillator. The digital count C obtained in this way is proportional to the temperature T. In fact, let us suppose that the frequencies of the two oscillators f_i can be expanded as:

$$f_i(T) = f_{i,0} \cdot [1 + TCf_i \cdot \Delta T] \tag{1.5}$$

where the two TCf_i are different and ΔT is the difference between the actual temperature and room temperature. Then, if the measurement lasts N periods of the first oscillator, the final digital count C will be equal to:

$$C = \frac{T_1}{T_2} \cdot N = \frac{f_2}{f_1} \cdot N = \frac{f_{2,0}}{f_{1,0}} \cdot \frac{1 + TCf_2 \cdot \Delta T}{1 + TCf_1 \cdot \Delta T} \cdot N \tag{1.6}$$

Since the relative change in frequency is typically small, up to a few percent at the boundaries of usual temperature ranges, (1.6) can be linearized as follows:

$$\begin{aligned} C &\approx \frac{f_{2,0}}{f_{1,0}} \cdot (1 + TCf_2 \cdot \Delta T) \cdot (1 + TCf_1 \cdot \Delta T) \cdot N \approx \\ &\approx \frac{f_{2,0}}{f_{1,0}} \cdot [1 + (TCf_2 - TCf_1) \cdot \Delta T] \cdot N \end{aligned} \tag{1.7}$$

Therefore, such a counter is indeed a temperature sensor with a sensitivity of $\frac{f_{2,0}}{f_{1,0}} \cdot (TCf_2 - TCf_1) \cdot N$, provided that the two resonators are at the same temperature.

This solution does not need an ADC, which is a big advantage in terms of design effort and power consumption. But, it requires an additional oscillator, with its own power consumption, that cannot be duty-cycled as

much as the ADC in the previous architecture. Moreover, the precision and accuracy of both sensing architecture must be assessed, as an error in the measurement of the temperature is directly translated into an error in the output frequency, which is the major specification of RTCs. This aspect will be analyzed more in detail in Section 2.4.

1.5 On the Use of Polysilicon Resonators for RTCs

On the basis of the discussion, the most promising option to make a MEMS RTC seems to combine a natively-compensated monocrystalline-silicon resonator to be then further compensated by a mixed-signal IC. This approach has led, for instance, SiTime Corp. to the first successful commercial application[18].

Nonetheless, in this Thesis the use of MEMS resonators realized in a polysilicon process will be investigated. This goal is motivated mainly by the *cost* and *repeatability* of polysilicon MEMS devices.

1.5.1 SOI vs Poly: Cost

The compensated highly-doped MEMS resonators must be realized in crystalline silicon, also called Single-Crystal Silicon (SCS). To fabricate such devices, a manufacturer starts from a bare Silicon-On-Insulator (SOI) wafer, where from a thin layer of crystalline silicon the whole mechanical structure is grown. The first difference between highly-doped SCS and standard polysilicon devices is therefore economical: the cost of a SOI wafer is about $300, whereas a wafer for a standard polysilicon process is about $30, ten times less.

Moreover, there are no wafers available off-the-shelf with the required doping level (around $1 \times 10^{20}\,\mathrm{cm}^{-3}$). Therefore, given the very large level of doping required, one should slowly grow a fraction of the structural layer thickness and dope it as required, and repeat these steps several times until the desired process height is reached. These additional fabrication steps can only increase the stark difference in production costs between the two solutions.

1.5.2 SOI vs Poly: Repeatability

It is common knowledge that doping levels are subject to a statistical spread. Thus, it is likely that devices from different wafers will have different properties, and even local fluctuations arise on a single wafer. Moreover, the

misalignment between the masks and the crystallographic axes is an additional source of variation, even though it is reported to be negligible [14].

No explicit comparison between the repeatability of polysilicon and doped SCS MEMS resonators can be found in the literature. Nonetheless, some data can be extrapolated from the work of Zaliasl *et al*. [17]. The measured devices are shown to have a peak-to-peak spread in their measured $f(T)$ curves of $\approx$80 ppm on a population of 28 000 samples. The entity of this spread is not compatible with the target $\Delta f/f$, therefore a five-point part-to-part calibration is used to reach the desired frequency accuracy.

One can suppose that if polysilicon resonators were used, the part-to-part spread on their $f(T)$ would be much smaller and a cheaper calibration routine would be needed. In fact, polysilicon resonators inherently do not suffer from the aforementioned causes of part-to-part spread.

This statement is treated more in detail in Chapter 2: it results that, indeed, polysilicon resonators show a good repeatability, thus potentially allowing a reduction in calibration costs. This intuition is the basis of this Thesis: the following chapters will analyze the feasibility of the idea, starting from the microelectromechanical device and moving then to the whole electronic system.

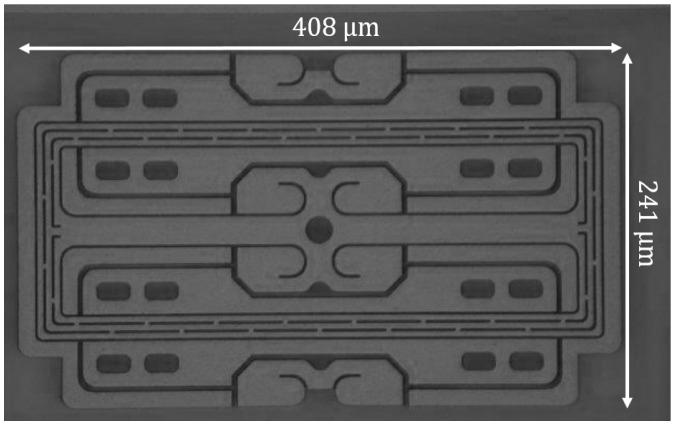
CHAPTER 2

Resonator Characterization

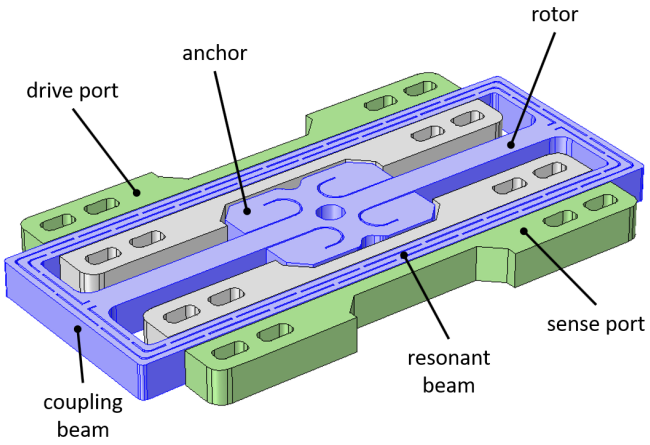
In this Chapter, the resonators used for the project are introduced. To produce a compensated RTC output, the $f(T)$ characteristic is measured and the achievable compensation accuracy is studied from a theoretical point of view. Particular emphasis is posed on the part-to-part spread of the employed polysilicon MEMS structures, leading to the hypothesis of a one-point only calibration to reach the target performances. The content of this Chapter has been published in [1], [2].

2.1 Description of the resonators

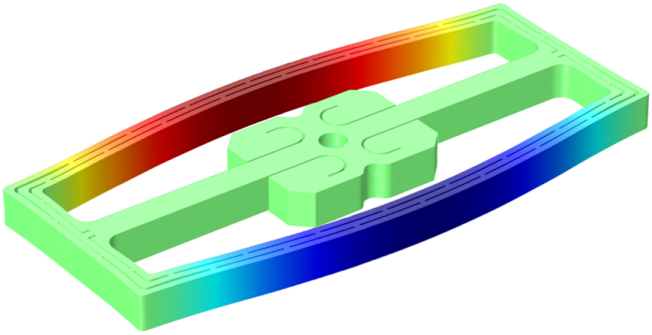
The resonators used for the project of this Thesis were dimensioned as described in this Chapter and then designed by STMicroelectronics in collaboration with the Dipartimento di Ingegneria Civile e Ambientale (DICA) of Politecnico di Milano: they are shown in Figure 2.1. The devices are fabricated with the 24 µm Thick Epitaxial Layer for Micro Accelerometers (ThELMA) process by STMicroelectronics [19]. The structural layer is grown from a standard epitaxy with n-type doping giving a square resistance of 15 $\Omega/\square$. A minimum in-plane gap of 0.8 µm by design (before



(a) *SEM image of the target resonator. the actuated slotted beams are* 351 µm *long and* 15 µm *wide.*



(b) *3D image taken from a COMSOL view highlighting the different functional parts of the resonator.*



(c) *Representation of the displacement in the mode of interest of the structure.*

Figure 2.1: *Images of the capacitive resonators designed in a* 20 µm *thick ThELMA process.*

any over-etch effect that bring the final gap to 1.2 µm[1]) and a maximum resonator footprint of 400 µm × 250 µm were chosen to cope with a final package footprint of 1 mm^2.

The structure, whose functional parts are highlighted in Figure 2.1b, is composed by two resonant beams that are actuated, and sensed, by two pairs of parallel-plate electrodes. The beams are then connected to the central anchor point by means of two coupling beams. More details can be found in [20].

2.1.1 Resonance Frequency

The choice of the resonance frequency f_0 of the MEMS device impacts many aspects at system level, therefore requiring particular care. By choosing the compensation strategy described in 1.4.2, the f_0 must be larger than the required output frequency f_{out} = 32.768 kHz, so that the fractional divider can divide f_0 by a factor between N and $N+1$ down to f_{out} while performing the compensation.

The value of N is determined by the maximum deviations of f_0 from its nominal value. These can be given by process spreads and by temperature changes. The smaller N, the larger the difference Δf between f_0/N and $f_0/(N+1)$, so the larger the range that can be compensated. With a −30 ppm/K 1st-order frequency drift, the maximum deviation due to temperature changes in the consumer range (300 K ±62 K) turns out to be ±1860 ppm. This is still much lower than process spreads on the frequency, which is dominated by etching non-uniformities and affects similarly polysilicon and SOI devices. This spread can be in the order of ±3 % (±30 000 ppm). This ±3 % spread then imposes an upper bound to N, regardless of the TCf of the process, expressed by:

$$\frac{f_0(1+0.03)}{N+1} = \frac{f_0(1-0.03)}{N} \Rightarrow N = 16 \tag{2.1}$$

A lower bound is posed by other limitations, because a lower N (i.e. a lower mechanical frequency):

- increases the part-to-part spread on the nominal frequency f_0, due to typically narrower springs.
- increases the effect of the electrostatic softening, due to a lower f_0 hence a typically lower mechanical stiffness.

[1]These devices were used in the first months of the project for the characterization activity that will be explained in the rest of the Chapter. Nonetheless, the devices that were coupled to the integrated circuit described in Chapter 4 featured a gap of 0.7 µm to reduce the motional resistance and, consequently, the power consumption of the oscillator.

- decreases the consumption of the sustaining oscillator, due to a lower f_0.
- increases the short-term output frequency variation (i.e. jitter), due to a larger Δf between periods resulting from different division factor.
- reduces the native jitter of the reference oscillator thanks to a typically better Q, even though this fact is negligible with respect to that expressed in the previous point.
- increases the part-to-part spread, due to a larger relative impact of the electrostatic stiffness on the lower total stiffness.

The programmable division factor N is thus set to 16/17, which implies that the nominal resonator frequency should equal $f_0 = f_{out} \cdot 16.5 = 540\,\text{kHz}$. Note that designing a high-frequency resonator has also the advantages of rejecting acceleration effects on the micromechanical structure. In fact, the displacement induced by an acceleration is proportional to the squared inverse of the resonance frequency.

2.1.2 Quality Factor and Transduction

The equivalent series resistance, or motional resistance R_m, has a direct impact on the power consumption of most low-power oscillator architectures [21], [22]. Therefore, it must be minimized to meet the consumption specification, which is a key requirement.

A generic expression for the R_m of a generic MEMS device is:

$$R_m = \frac{b}{\eta^2} = \frac{\omega_0 m}{Q} \cdot \frac{1}{\eta^2} \tag{2.2}$$

where $\omega_0 = 2\pi f_0$ is the resonant angular frequency, m the modal mass, Q the quality factor, η the transduction factor between the electrical and mechanical domains. Considering ω_0, k and m fixed by the choice of N, actions were taken to maximize Q and η.

The Q optimization involves minimizing all sources of energy loss during the operation of the resonator. Therefore, the pressure inside the MEMS package was minimized to a level below 50 µbar, to make the energy loss due to the fluid damping negligible. Moreover, the moving structure is anchored to the silicon substrate at nodes of the desired mode of operation.

The main mechanism of energy loss that is left after these precautions is the thermo-elastic damping. This source of loss is given by the heat flow that develops between portions of the moving rotor. In fact, some areas of

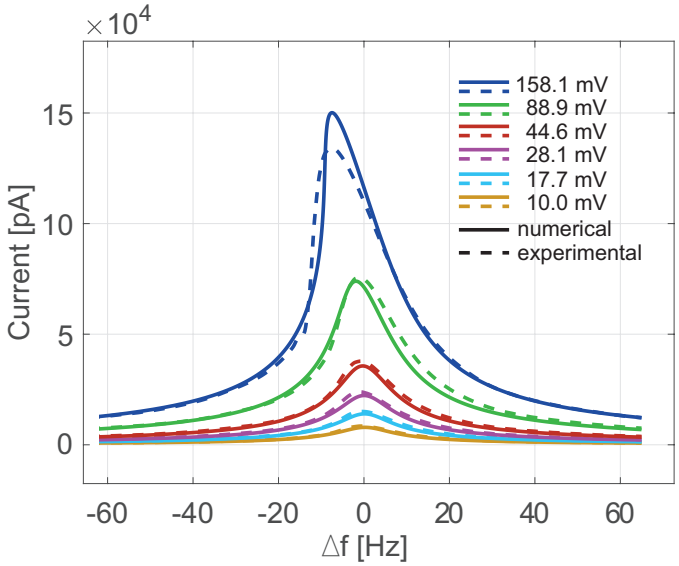


Figure 2.2: *Measured output current of the MEMS resonators with varying driving amplitude: the observed non-linear behavior is in agreement with models present in the literature.*

the beam are subject to contraction which increases the local temperature, whereas other areas expand and decrease their local temperature. Therefore, slots were introduced to hinder the heat flow between hot and cold areas of the resonator, reaching a Q in the order of 45 000.

The factor η is maximized with the use of a transduction mechanism similar to a parallel-plate actuation, more efficient than a comb-finger actuation, and with the obvious choice of the minimum available gap.

2.1.3 Non-Linearity Modeling

Since large oscillation amplitudes are usually desirable in an oscillator, the non-linearities of these MEMS structures were studied. In fact, the larger the driving amplitude applied to the device, the more the shape of the transfer function is deformed and the resonant frequency shifts away, in our case to the left due to the electrostatic softening. With the use of a FEM reduced-order model like the one described in [23], the effect of the non-linearities was predicted as shown in Figure 2.2. These results are in agreement with the experimental observations. Therefore, to not enter a working region where the MEMS frequency becomes too sensitive to the driving voltage amplitude, all the measurements were carried out by

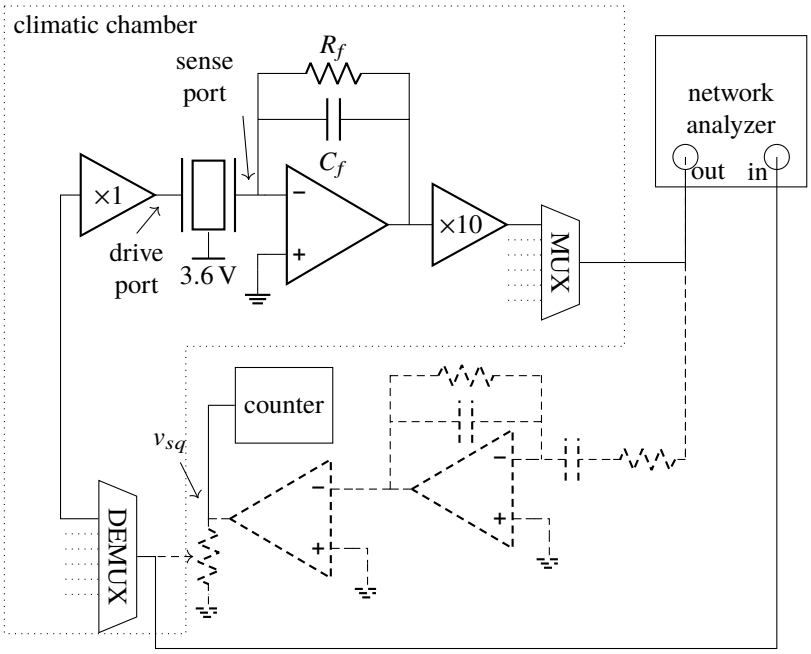


Figure 2.3: *Sketch of the setup used to characterize the resonators.*

considering a maximum excitation voltage of ≈80 mV.

2.2 Characterization Setup

The primary goal of the characterization work is to gather data regarding the $f(T)$ dependence of the MEMS devices. Additionally, the measurements should help the mechanical designers validate their simulation results.

Therefore, a custom setup was built to accommodate all these needs, that is sketched in Figure 2.3.

2.2.1 Open-Loop Characterization

The resonators are interfaced to the custom board with a CLCC68 ceramic carrier on which they are wire-bonded. The drive signal is injected into the drive port and the resulting motional current is read with a charge amplifier whose feedback capacitance and resistance are set to 500 fF and 1 GΩ respectively. The sensed signal is then further amplified. The transfer function of the whole chain is measured with the use of an HP4195A Network Analyzer who drives the chain and reads the resulting output. All the dashed stages of Figure 2.3 are disconnected in this operating mode.

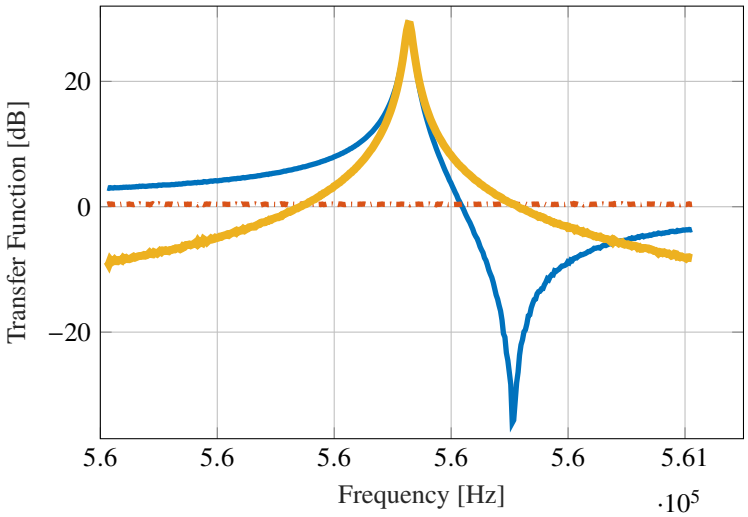


Figure 2.4: *(blue) Transfer function modulus between the MEMS drive terminal and the voltage output of the read-out electronics with rotor voltage at* 3.6 V*, (orange dash)-dotted with rotor voltage at* 0 V*, (yellow) MEMS transfer function subtracted of the effect of the feedthrough capacitance.*

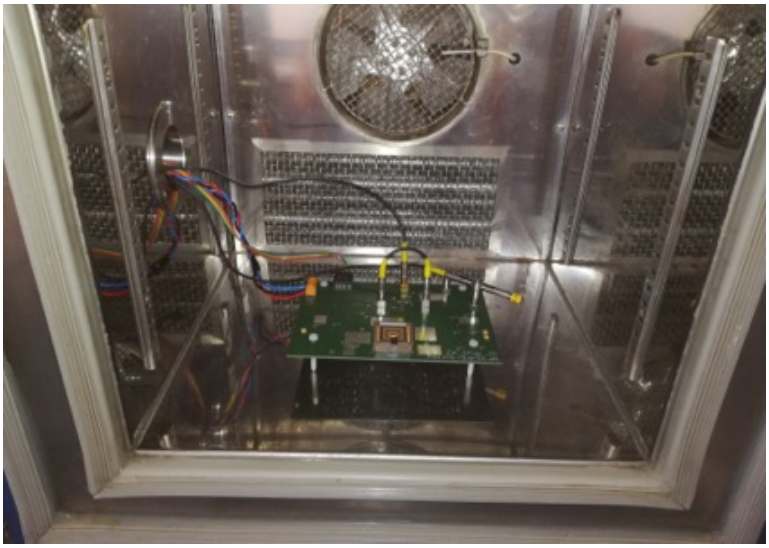


Figure 2.5: *Setup inside the climatic chamber.*

On a single MEMS die four different mechanical variants were present to validate the design choices. Specifically, the four designs are similar but differ in the presence/absence of the anti-thermoelastic-damping slots, in the nominal resonant frequency and in the anchor design. Therefore, a multiplexer and a demultiplexer were added to test with four different chains the four different MEMS resonators.

The board can then be placed inside a climatic chamber, as shown in Figure 2.5. The chamber senses its temperature with a sensor that is inevitably quite far from the devices. Therefore, to measure the temperature of the resonators as accurately as possible, the board was equipped with a LM35 temperature sensor that was placed close to the ceramic carrier.

Parameter Extraction

The Transfer Function (TF) of the considered resonators is that of a series RLC circuit with a feedthrough capacitance in parallel.

To extract the relevant parameters while getting rid of the spurious feedthrough capacitance, the TF of the MEMS devices are measured both with a rotor voltage applied – 3.6 V typically – and with 0 V applied. An example of this is given by the blue solid and orange dash-dotted traces of Figure 2.4 (only the modulus is shown). In this way, one can mathematically subtract the two complex waveforms and obtain the pure RLC series TF (yellow), from which the MEMS parameters can be extracted in an easier and more accurate way.

In fact, an automatic MATLAB routine extracts the f_0 as the frequency of the peak of the TF and the motional resistance R_m as the peak value divided by the known trans-impedance of the whole read-out chain. The Q is estimated from the slope of the phase of the TF close to f_0. The motional capacitance and inductance C_m and L_m are then determined analytically as $C_m = \frac{1}{\omega_0 R_m Q}$ and $L_m = \frac{1}{\omega_0^2 C_m}$.

Estimation of the Setup Error

Errors on both the measurement of f_0 and T have an impact on the precision with which the TCf can be estimated. Therefore, the impact of these errors will be described and quantified.

The error committed on the extraction of the resonance frequency of the MEMS structure was made negligible by choosing a sufficiently small frequency range around the f_0 and by subsequently using a *spline* interpolation on the acquired data embedded in the MATLAB extraction routine. This error amounted to about 1 ppm for each f_0 datum.

The most significant error limiting the accuracy and resolution of the measurement was the uncertainty about the temperature of the MEMS device. There are many causes for this error. One is the good, but not perfect, thermal homogeneity between the sensor and the device. This problem was exacerbated by the residual temperature fluctuation due to the temperature controller of the chamber, particularly in the lower part of the temperature range. Another cause is the resolution and inaccuracy of the used temperature sensor, which limited the temperature readout to an estimated 0.15 °C uncertainty (1σ).

Translating the temperature error into an estimated frequency error for better comparison, all of this resulted in an uncertainty f_ϵ ($1\ \sigma$) on each f_0 sample given by:

$$f_\epsilon = T_\epsilon \cdot \frac{\partial f_0}{\partial T} \approx 0.15\,\mathrm{K} \cdot 30\,\mathrm{ppm/K} = 4.5\,\mathrm{ppm} \tag{2.3}$$

where T_ϵ indicates the total estimated temperature error.

The error committed on the estimate of the TCf by the proposed method is computed *a posteriori*, as a 1-σ dispersion with the following formula:

$$TCf_\epsilon = \sqrt{\frac{\sigma_{res}^2}{\sum_i T_i^2}} = 0.3\,\mathrm{ppm/K} \tag{2.4}$$

where T_i indicates the temperature data and σ_{res}^2 the variance of the residuals on a representative number of measurements. This formula is obtained from the standard linear regression theory [24], where the residuals are defined as the difference between the fitted data and the actual data. This estimate also does not take into account the larger fluctuations in temperature at the lower bound of the considered range, so that this estimate can be considered smaller than the real accuracy.

2.2.2 Closed-Loop Characterization

The board was designed to operate as a discrete-component oscillator by connecting also the dashed stages of Figure 2.3. They consist in an integrator that provides the required 90° shift to satisfy the Barkhausen criterion and a hard limiter that compresses the harmonic gain of the whole loop to 1. The driving voltage of the MEMS can be tuned with a potentiometer and was typically set to 50 mV to keep the device within its linear region. All the singularities were placed at least two decades far from the expected oscillation frequency, to avoid unnecessary phase shifts.

Finally, the oscillation frequency is measured with an Agilent 53131A frequency counter at the node indicated as v_{out}.

	f_0 [kHz]	R_m [kΩ]	Q [-]	L_m [kH]	C_m [aF]
mean	558.86	589.56	46 240	7.77	10.47
std	0.89	27.22	412	0.43	0.54

Table 2.1: *Electromechanical parameters of the MEMS resonators estimated from the transfer function measurements.*

2.3 Measurement Results

The f_0, R_m, Q, C_m and L_m parameters extracted from the measurements are reported in Table 2.1. The main difference with respect to what was expected is a shift in the f_0 of the devices, that was targeted at 541 kHz because of the reasons pointed out in Section 2.1.1. This is due to the use of a new machine at the foundry for which there was no expertise regarding the control of the over-etch at the time of the fabrication of the devices.

On the other hand, the slotting along the moving beam was confirmed to be very effective in improving the energy losses due to thermo-elastic damping, raising the Q by a factor of 4.

Since the values of R_m have been considered too large to achieve a satisfactory consumption of the integrated oscillator, the MEMS will be re-designed with tighter gaps of 0.7 µm and fabricated with a new etching machine, compatible with such gaps.

2.3.1 TCf Measurement

To measure the $f(T)$ dependence, the same TFs were measured with the board placed inside the climatic chamber. The measurements were taken from 5 °C to 85 °C at strides of 10 °C, waiting about 20 min between consecutive measurements to cancel any difference in thermal inertia between the MEMS die and the temperature sensor.

Figure 2.6 shows the result of this characterization for all the different MEMS designs for a population of 44 devices (11 dies with 4 different resonators each). As expected, the TCf is dominated by a ≈ -30 ppm/K linear term that derives from the variation in temperature of the Young modulus of silicon of −60 ppm/K. Also, it is to be noted that the frequency spread due to the fabrication process – or *frequency offset* – is larger than the variations of frequency caused by the temperature. This fact would be even more evident if a population of thousands of devices was measured.

Figure 2.7 shows the TCfs obtained from the $f(T)$ traces fitted with a 1st-order linear regression. The fitted TCf values have a mean value of ≈ -30.9 ppm/K and show no significant dependence on the specific

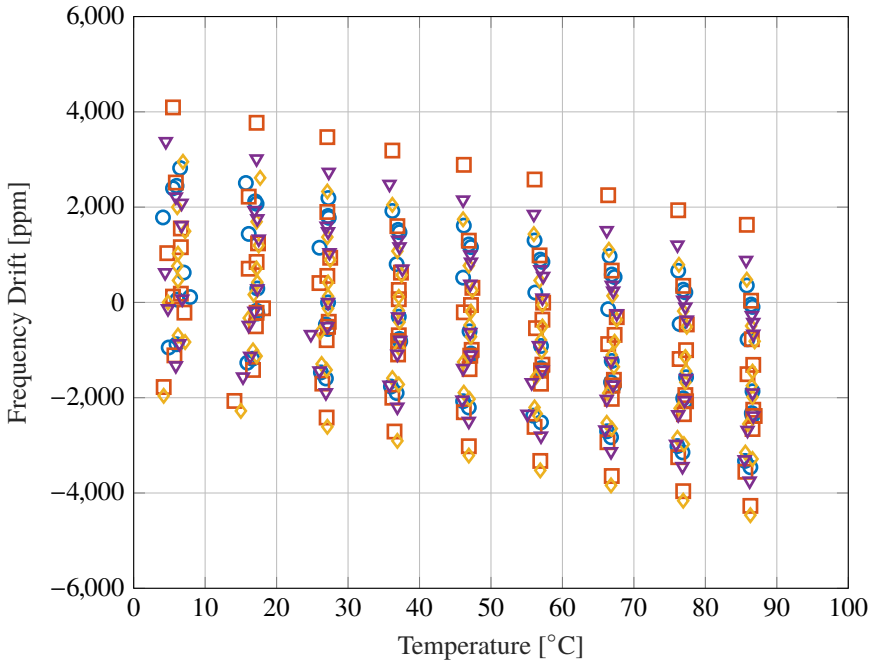


Figure 2.6: *Result of the characterization of the TC f for all the different MEMS structures included in the MEMS die. Each color/marker indicates a unique design. All the $f(T)$ traces have been transformed into* ppm *with a normalization to the* average *(between samples of the same design) f_0 at room temperature to emphasize the existing spread of the nominal frequency across the population.*

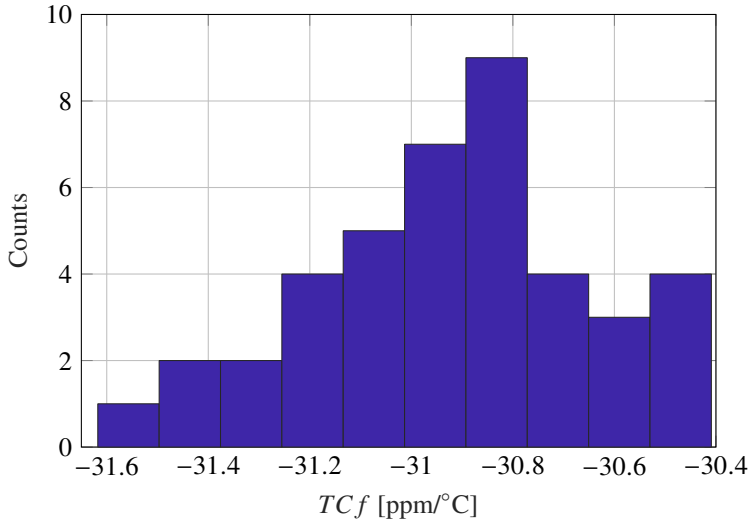


Figure 2.7: *Histogram of the obtained TCf values, without distinguishing between the different variations over the target design. The average value is* $-30.9\,\mathrm{ppm/K}$ *and the real dispersion of the devices is likely to be hidden under an estimated setup error of* $0.3\,\mathrm{ppm/K}$.

design, regardless of their difference in f_0, Q, anchoring and actuation strategies. The dispersion of these values is compatible with the setup accuracy estimated as $\sigma = 0.3\,\mathrm{ppm/K}$ in Section 2.2.1.

2.3.2 Compensation and Residual Temperature Dependence

It is interesting to elaborate these data to understand the limits of an electronic compensation. To do so, a 1-point calibration can be simulated in the following way:

1. The TCf for the population is computed as the average of the fitted TCf for the single device.

2. A straight line with the *average* TCf is subtracted to the single $f(T)$ traces. This mimics a family compensation, where a small portion of devices are characterized with time-consuming operations and these data are then applied to the whole population.

3. The bias of each resulting $\Delta f(T)$ trace is removed from the trace itself. This mimics the calibration of each device by measuring its f_0 at a reference frequency.

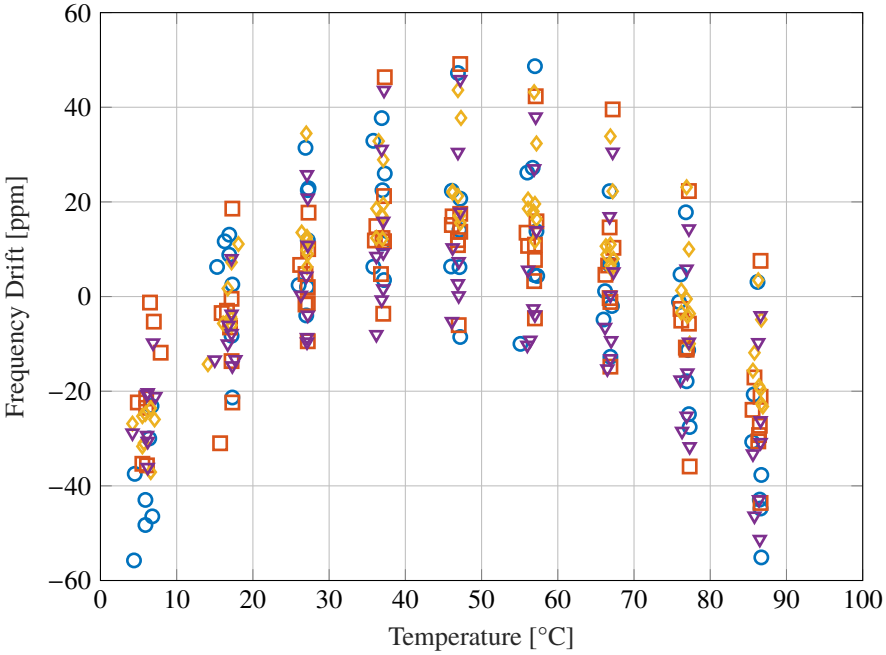


Figure 2.8: *Residual frequency drift after an* average *1st-order compensation.*

This operation leads us to the 1st-order residuals that are shown in Figure 2.8. The 2nd-order Temperature Coefficient of Frequency TCf_2 is highlighted. This higher-order temperature dependence is clearly much smaller than the linear one, but still gives a clear residual frequency drift of ≈50 ppm, too large for the target requirements of Table 1.1. This becomes ≈ ± 50 ppm considering also the spread between measurements.

Therefore, it is natural to apply the same procedure with a 2nd-order compensating polynomial computed as the average of the 2nd-order fitted $f(T)$ traces. The obtained 2nd-order residuals are shown in Figure 2.9; the compensating polynomial is the following:

$$\frac{\Delta f}{f}(T) = OS - 28.64\,\text{ppm}/^{\circ}\text{C} \cdot T - 0.024\,\text{ppm}/^{\circ}\text{C}^2 \cdot T^2 \tag{2.5}$$

where T is the temperature in degrees Celsius and OS is an offset that is computed for each trace.

In this case, no significant trend can be highlighted with a dispersion bounded roughly at ±20 ppm. This limit is somewhat compatible with the setup accuracy estimated before.

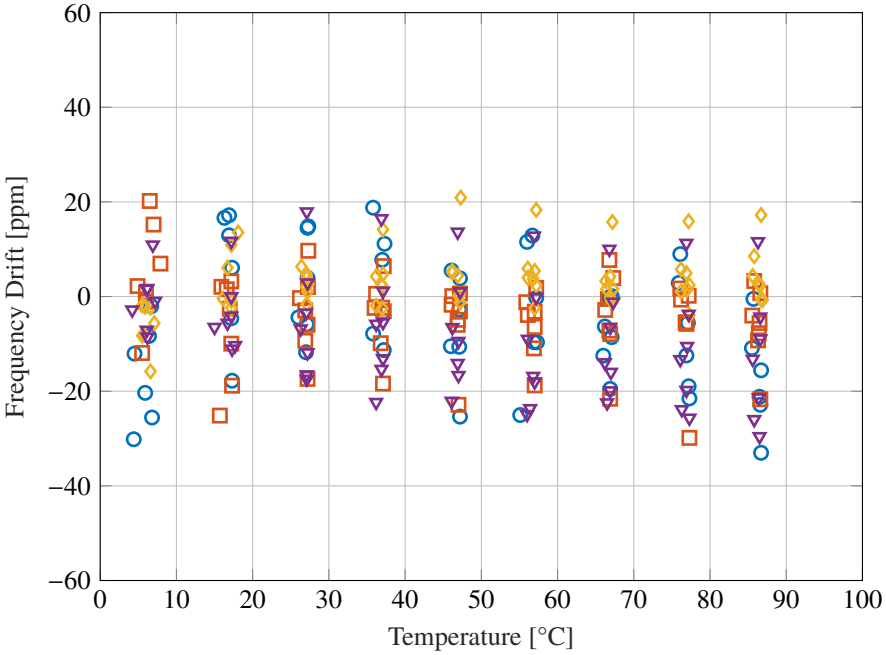


Figure 2.9: *Residual frequency drift after a 2^{nd}-order compensation.*

Verification Dataset

The measurement analysis exposed up to now has found the coefficients for the compensation from the same devices on which the mathematical compensation is applied. Therefore, it is interesting to see the result of the application of a compensating polynomial found from a training dataset applied to devices from a different verification dataset. Moreover, to highlight any possible bias in the measurement methodology, the verification dataset was obtained by inserting the MEMS devices within the PCB oscillator described in Section 2.2.2. Only the oscillating frequencies were measured, without estimating the resonant frequency from the TF of the resonator.

The result of this experiment is shown in Figure 2.10. It is evident that, although these devices were not used to compute the average $f(T)$ dependence, the compensation is good enough to reach the same level of performance.

2.4 Measurement Interpretation

The theoretical compensation and the application to the verification dataset show a level of repeatability of these devices up to ±20 ppm, limited by the

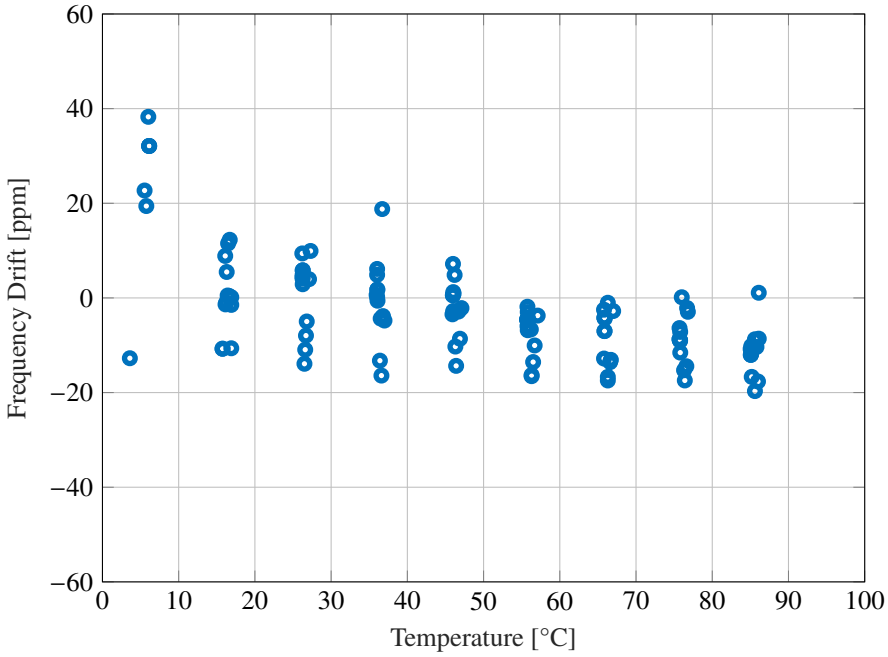


Figure 2.10: *Frequency drift in temperature of the devices from the verification dataset. The residual spread of the traces has seen an improvement with respect to the experiment of Figure 2.9 because of a new automatic routine that made possible to achieve a better stability of the temperature inside the climatic chamber.*

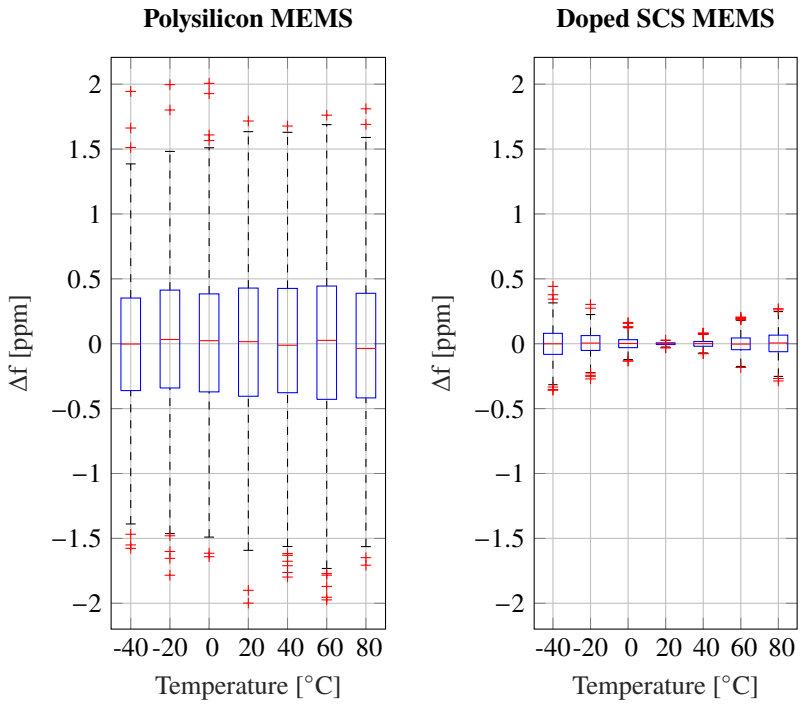


Figure 2.11: *Frequency accuracy of a RTC based on a polysilicon MEMS (left) and a highly-doped SCS MEMS (right) with no part-to-part spread but with a simulated* $\pm 3\sigma$ *inaccuracy of the temperature sensor of* $\pm 0.12\,^{\circ}\mathrm{C}$ *on 1000 samples. The red line in the middle of the box represents the average value for that specific temperature value. The boxes are bounded at the* 1^{st} *and* 3^{rd} *quartiles, whereas the lines stretch as far as* $\pm 3\sigma$.

resolution of the setup.

This fact gives credibility to the claim of Section 1.5 that the use of polysilicon resonators for RTCs can be a cheaper approach in terms of production and calibration costs than more widespread solutions adopting custom processes and multi-point calibration on each device. In this case, the fabricated RTC would just require a single-point calibration to compensate for the frequency offset, i.e. the spread in the initial frequency of the resonator induced by process non-uniformities.

A possible objection is that even though the described resonators are sufficiently repeatable, this may not hold true for the coupled temperature sensor embedded in the coupled ASIC.

Indeed, the use of a resonator with a poorer frequency accuracy puts more strain on the rest of the system, temperature sensor included. In fact, any error on the sensed temperature is translated into an output frequency

error with a sensitivity equal to the TCf. Therefore, it becomes mandatory to study the repeatability of temperature sensors to verify the plausibility of the use of one-point calibration polysilicon MEMS resonators for RTCs.

As emerges from a comprehensive survey by Makinwa [25], a good state-of-the-art reference is the BJT-based temperature sensor published in [26]. The paper reports a sensor that, by making use of Dynamic Element Matching (DEM) techniques, a batch calibration and an individual single point trim, reaches an inaccuracy of $\pm 120\,\mathrm{mK}$ ($\pm 3\sigma$) measured across 80 samples. This is achieved in a $0.16\,\mu\mathrm{m}$ CMOS process at a $0.16\,\mathrm{mm}^2$ area.

A simulation was carried out to simulate the effect of this level of inaccuracy in the temperature sensor on a perfectly repeatable resonator. The inaccuracy for each point in temperature was considered an independent Gaussian variable with zero-mean and standard deviation $\sigma = 120\,\mathrm{mK}/3 = 40\,\mathrm{mK}$. The result is reported in Figure 2.11 for both a high-TCf polysilicon device and a low-TCf one.

What emerges is that, indeed, the same temperature sensor produces a less accurate output frequency if a $-30\,\mathrm{ppm/K}$ polysilicon resonator is used. Nonetheless, state-of-the-art sensors are able to to compensate this reference frequency down to few parts per million – $\pm 2\,\mathrm{ppm}$ was found in this simulation – which is compatible with the specifications stated in Table 1.1.

CHAPTER 3

System Requirements

This Chapter analyzes in detail the compensation scheme briefly described in Chapter 1: the constraints posed by the application are studied, the feasibility of the scheme is proved and the main system-level parameters are fixed. Moreover, a novel solution for jitter suppression is devised: its implementation has been the object of a patent [4]. Some portions of this Chapter are published in [1], [3].

3.1 System Description

As briefly described in Section 1.4.2, a popular way to compensate at system-level the thermal drift of an oscillator is the use of the architecture re-shown in Figure 3.1. Here, to obtain a clock signal at frequency $f_{rtc} = 32.768\,\text{kHz}$, an oscillator, based on the MEMS resonator, running at a higher frequency f_0 is used. This oscillating signal is characterized by a relatively clean phase spectrum.

To achieve the correct output frequency f_{rtc}, a multi-modulus divider – whose modulus is switched between two consecutive integer values N and $N + 1$ by the Modulus Control (MC) input – is used to divide its f_0 input

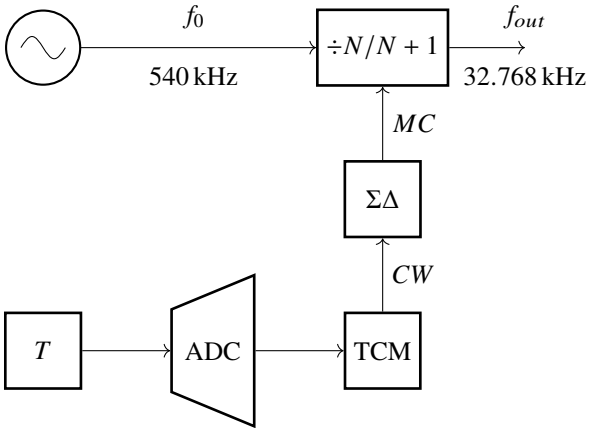


Figure 3.1: *Sketch of the used compensation scheme.*

frequency, so that:

$$f_0 = f_{rtc} \cdot N_{frac} \tag{3.1}$$

with

$$N < N_{frac} < N + 1 \tag{3.2}$$

The MC signal is set by the temperature sensing chain. A temperature sensor acquires the current temperature and digitizes it with an Analog-to-Digital Converter (ADC). Since the temperature variations are typically slow, the temperature is sensed only up to a few times per second, with a rate f_s. A digital block, Temperature Compensation Machine (TCM), maps the digitized temperature to a multi-bit signal called Command Word (CW). The CW represents, on B bits, the fractional part of the desired division factor N_{frac}. In formulas:

$$N_{frac} = N + \frac{CW}{2^B}, \quad 0 \leq CW < 2^B \tag{3.3}$$

The translation between the digital word CW and the fractional division factor is made possible by a Digital ΔΣ Modulator (DDSM) that encodes a fixed (or, better, slowly-varying) multi-bit word CW into a single-bit signal MC whose average value represents the value $CW/2^B$. In this way, the output of the multi-modulus divider is a squarewave whose period is constantly changed between $N \cdot T_0$ and $(N + 1) \cdot T_0$ but whose average frequency is the desired 32.768 kHz.

Nevertheless, the fractional division achieved thanks to the ΣΔ modulation introduces a jitter that can be very large. In the worst case when $N_{frac} = N + 0.5$, this jitter has a constant value, with alternating sign, of

Symbol	Comment
f_0	Frequency of the reference oscillator
T_0	Period of the reference oscillator
f_{rtc}	32.768 kHz output frequency
T_{rtc}	Output period
$f_{out}(T)$	Output frequency, after compensation
$N \quad (N+1)$	Minimum (maximum) division modulus of the divider
N_{frac}	Fractional division modulus employed
f_s	Temperature sampling rate
DDSM	Digital $\Delta\Sigma$ Modulator
CW	Command Word, input of the DDSM
B	Number of bits of the DDSM
α	Linear Temperature Coefficient of Frequency
β	Quadratic Temperature Coefficient of Frequency
$\overline{\alpha}$	Measured Linear Temperature Coefficient of Frequency
$\overline{\beta}$	Measured Quadratic Temperature Coefficient of Frequency
ΔT	Difference between actual and calibration temperature
ΔT_m	Measured difference between actual and calibration temperature
T_{os}	Offset of the temperature sensor, constant in temperature
T_{ia}	Inaccuracy of the temperature sensor, varies with temperature
T_ϵ	Calibration temperature error
T'_R	Calibration temperature as measured by the sensor of each part
f'_0	f_0 of the specific device as measured in the calibration phase

Table 3.1: *Table of symbols used in the study of the compensation system.*

$T_0/2$, much larger than the native jitter of the reference oscillator that is a fraction of this value. Because of this problem, in applications where a small jitter is required, a phase filter can be introduced after the divider. This block will be analyzed more in detail in Section 3.4.

To summarize, a list of the symbols used in this Chapter and in the following is available in Table 3.1.

3.2 Choice of Parameters

The first thing to consider when sizing the system are the two available division factors $N/N+1$ and the number of bits B of the DDSM. These two parameters limit the compensation of the frequency drift with non-idealities and errors that must be kept within the specification. A further degree of freedom is the order of the $\Sigma\Delta$ modulator, that will be set to 1[1].

[1]A modulation order larger than 1 would increase the order of the frequency noise shaping, allowing an easier filtering with a low-pass filter. Nevertheless, the strategy used in this Thesis to suppress the output jitter, exposed later in the Chapter, does not benefit from a higher-order shaping. An order of 1 is then chosen to use less

The output frequency f_{out} drift is well-described by the following law[2]:

$$f_{out}(T) = f_0 \cdot \left[1 + \alpha \cdot (T - T_R) + \beta \cdot (T - T_R)^2\right] \cdot \frac{1}{N_{frac}} \tag{3.4}$$

where f_0 is the native MEMS frequency at a reference temperature T_R which is affected by process spreads, α and β are the 1st-order and 2nd-order Temperature Coefficients of Frequency and T is the temperature of the MEMS resonator.

To have a compensated RTC output, N_{frac} must be adjusted respecting the condition on the output frequency $f_{out}(T) = f_{rtc} = 32.768\,\text{kHz}$ which, combined with (3.4), gives:

$$N_{frac}(T) = \frac{f_0}{f_{rtc}} \cdot \left[1 + \alpha \cdot (T - T_R) + \beta \cdot (T - T_R)^2\right] \tag{3.5}$$

More practically, adjusting the average division factor N_{frac} requires a precise mapping, operated by the TCM, between the MEMS temperature T and the Command Word CW, which is obtained combining (3.5) and (3.3):

$$CW = 2^B \cdot \left[N_{frac}(T_m) - N\right] = \tag{3.6}$$

$$= 2^B \left\{\frac{f_0}{f_{rtc}} \left[1 + \overline{\alpha} \cdot (T_m - T_R) + \overline{\beta} \cdot (T_m - T_R)^2\right] - N\right\} \tag{3.7}$$

Here coefficients $\overline{\alpha}$ and $\overline{\beta}$ are introduced, to mark the difference between the actual α and β, that can have a variation between the various parts, and the average coefficients $\overline{\alpha}$ and $\overline{\beta}$ measured and stored on chip. Also, T_m is introduced to indicate the *measured* temperature, affected by errors such as noise, offset...

Combining (3.4) with (3.7), it is possible to obtain the final output frequency as a function of the temperature only:

$$f_{out}(T) = \frac{f_0(T)}{N_{frac}(T)} = \tag{3.8}$$

$$= \frac{f_{rtc} \cdot \left[1 + \alpha \cdot \Delta T + \beta \cdot \Delta T^2\right]}{1 + \overline{\alpha} \cdot \Delta T_m + \overline{\beta} \cdot \Delta T_m^2 + \frac{f_{rtc}}{f_0} \cdot \frac{\epsilon_q}{2^B}} \tag{3.9}$$

where a stochastic term ϵ_q ($|\epsilon_q| < 0.5$) has been added to consider the quantization error caused by a finite B, and ΔT_m indicates the *measured*

hardware and save some power.

[2] In this Chapter the factors TCf_1 and TCf_2 are called α and β for visual brevity.

temperature difference with respect to the *measured* room temperature. Since the deviation from the nominal frequency is small – thousands of ppm at most – one can linearize this expression, obtaining a more manageable expression:

$$\begin{aligned}\frac{f_{out}(T)}{f_{rtc}} \approx &\left(1+\alpha\cdot\Delta T+\beta\cdot\Delta T^2\right)\cdot\\ &\cdot\left(1-\overline{\alpha}\cdot\Delta T_m-\overline{\beta}\cdot\Delta T_m^2-\frac{\epsilon_q}{2^B}\right)\approx\\ \approx &1+\alpha\cdot\Delta T-\overline{\alpha}\cdot\Delta T_m+\\ &+\beta\cdot\Delta T^2-\overline{\beta}\cdot\Delta T_m^2-\frac{f_{rtc}}{f_0}\cdot\frac{\epsilon_q}{2^B}\end{aligned}\tag{3.10}$$

From (3.10), it is possible to highlight the possible causes of errors in the output frequency that must be kept within the $\Delta f/f$ specification:

- an imperfect knowledge of the device coefficients α and β.
- an error in the measured temperature ΔT_m – which can be due to noise, offset, or inaccuracy of the sensor – that causes a wrong CW to be used.
- the quantization of N_{frac} expressed by the ϵ_q term.

To stay within the $\Delta f/f$ requirement, all these contributions must be kept below the maximum allowed level. In the next sections of this Chapter all the contributions will be analyzed, leading to a system-level sizing of the architecture.

3.2.1 Division Factor and Reference Frequency

Considering the output frequency fixed at f_{rtc}, the choice of N also fixes the reference frequency f_0. A first boundary for this sizing is given by the spread on the resonant frequency of the MEMS. In fact, N must be chosen so that all the fabricated devices can be compensated. In other words, the maximum and minimum frequency of the whole distribution of f_0 must lie within the range of possible compensation, i.e. $f_{min} > f_{rtc}\cdot N$ and $f_{max} < f_{rtc}\cdot(N+1)$.

Given a percentage spread on the MEMS frequency ϵ, that according to the industrial partner can reasonably be close to $\pm 2\,\%$, one can write as

follows:

$$\epsilon < \frac{f_{max} - f_{min}}{\frac{f_{max} + f_{min}}{2}} = \frac{\frac{f_0}{N} - \frac{f_0}{N+1}}{\frac{f_0}{N+1/2}} \tag{3.11}$$

In the inequality, f_0 simplifies so that N is the only relevant parameter to guarantee that all the devices can be compensated. Developing the computation, one arrives to the following condition:

$$\epsilon < \frac{1}{N + \frac{1}{2}} \Rightarrow N < \frac{1}{\epsilon} - \frac{1}{2} = 24.5 \tag{3.12}$$

The condition expressed by (3.12) also imposes an implicit constraint on the MEMS resonant frequency f_0. In fact, if N cannot be larger than 24, the maximum target MEMS frequency $f_{0,max}$ is:

$$f_{0,max} = f_{rtc} \cdot 23.5 = 770\,\text{kHz} \tag{3.13}$$

Of course, one could resort to an f_0 larger than $f_{0,max}$ and use a divider whose modulus can switch between N and, e.g., $N + 2$. Nevertheless, even though it would be a working choice, this solution is not optimal: the power consumption of the oscillator is proportional to the square of its oscillation frequency, as will be highlighted in Chapter 4.

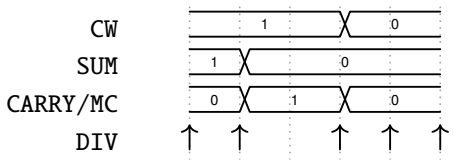
3.2.2 Compensation Bits and Temperature Sampling Frequency

Once N is decided, the FSR of the compensation, i.e. the maximum achievable division factor, is fixed. The other key factor in describing the technique is the choice of the granularity, i.e. the Least Significant Bit (LSB). This aspect depends only on the number of bits B used for CW and the DDSM. In fact, the more bits are used to describe CW, the finer the achievable division factor is. In formulas, the residual frequency drift $\left.\frac{\Delta f}{f}\right|_q$ that we expect due to the quantization caused by the finite B will be equal to the full-scale modulus variation divided by 2^B:

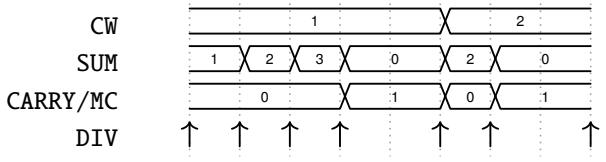
$$\left.\frac{\Delta f}{f}\right|_q = \frac{1}{N + \frac{1}{2}} \cdot \frac{1}{2^B} \tag{3.14}$$

By imposing that $\left.\frac{\Delta f}{f}\right|_q < \frac{\Delta f}{f}$, the condition on the minimum B is obtained:

$$\frac{1}{N + \frac{1}{2}} \cdot \frac{1}{2^B} < \frac{\Delta f}{f} \Rightarrow B > \log_2\left(\frac{1}{N + \frac{1}{2}} \cdot \frac{1}{\frac{\Delta f}{f}}\right) \tag{3.15}$$



(a) *1 bit case: in the worst case $CW = 1$, the 0.5 average value is encoded in two periods of DIV.*



(b) *2 bit case: in the worst case $CW = 1$ (or $CW = 3$), one needs to wait 4 periods of DIV before the average output value reaches 0.25 (or 0.75).*

Figure 3.2: *The timing diagram shows two simple examples of the evolution of the main signals involved in the DDSM operation. The larger the number of bits B, the longer one has to wait for the correct conversion of the Command Word CW into the average value of the single-bit Modulus Control MC.*

Nonetheless, one can not rely on using an arbitrary large B to achieve a resolution as large as desired. In fact, the value of B is in trade-off with the time needed to wait that the average output of the DDSM – the MC signal – reaches the correct value. This aspect is detailed in two simple cases in Figure 3.2. In Figure 3.2a, the evolution of the divided clock DIV, the CW, SUM and CARRY/MC waveforms are visualized assuming $B = 1$. In this case only divisions by N and $N + 0.5$ are possible and the worst-case conversion time T_{conv} is equal to $2T_{rtc}$, which is the duration of the periodic cycle that characterizes the evolution of the DDSM state.

The case when $B = 2$ is represented, instead, in Figure 3.2b. When $CW = 1$, the implemented division factor is $N + 0.25$, and now the system waits for 4 periods of DIV before the desired resolution is achieved. In other words, T_{conv} increases by a factor of 2 for each bit added, a fact intrinsic in the $\Sigma\Delta$ technique.

To summarize this example, it is possible to write in formulas that T_{conv} is equal to:

$$T_{conv} = T_{rtc} \cdot 2^B \tag{3.16}$$

or an integer multiple of this value. Since the temperature is sampled relatively infrequently, some time is available to apply the correct compensation MC for each temperature sample. Nevertheless, the maximum tolerable temperature gradient $\frac{dT}{dt}\Big|_{max}$ imposes a lower boundary to the min-

imum temperature sampling frequency f_s to keep the difference between the measured temperature T_m and the actual temperature T under control and the frequency drift within the specification. Expressing the temperature error as $\frac{dT}{dt}\Big|_{max} \cdot \frac{1}{f_s}$, one can write this condition as follows:

$$\frac{dT}{dt}\bigg|_{max} \cdot \frac{1}{f_s} \cdot \alpha < \frac{\Delta f}{f} \tag{3.17}$$

From this requirement, imposing the necessary condition $T_{conv} < \frac{1}{f_s}$ and making use of (3.16), it is possible to find the maximum number of bits that can be effectively encoded within the available time:

$$T_{rtc} \cdot 2^B < \frac{\Delta f}{f} \cdot \frac{1}{\alpha \cdot \frac{dT}{dt}\Big|_{max}} \Rightarrow B < \log_2 \left(\frac{\Delta f}{f} \cdot \frac{1}{\frac{dT}{dt}\Big|_{max} \cdot \alpha \cdot T_{rtc}} \right) \tag{3.18}$$

By combining (3.15) and (3.18), it is possible to summarize the conditions that B must respect, which constitutes an upper and a lower boundary:

$$\log_2 \left(\frac{1}{N + \frac{1}{2}} \cdot \frac{1}{\frac{\Delta f}{f}} \right) < B < \log_2 \left(\frac{\Delta f}{f} \cdot \frac{1}{\frac{dT}{dt}\Big|_{max} \cdot \alpha \cdot T_{rtc}} \right) \tag{3.19}$$

The three conditions imposed on N and B expressed in (3.12) and (3.19) can be plotted on a design-space plane with N and B as variables, where each of the mentioned equations prohibits a portion of the infinite (N, B) pairs. The result is shown in Figure 3.3, where the white region represent the allowed subset of (N, B) pairs.

The number of available options is not huge, but still there is room to analyze the possible choices and make an optimal decision. To begin with, it makes no sense to work close to the green/upper boundary, close to the green region, once N is fixed. The number of bits would encode a granularity in the compensation much smaller than the required residual drift $\Delta f / f$. At fixed N, a non-minimum B would stress the resolution of the temperature sensor. This would cause an increase in the consumption of the temperature sensor.

As a consequence, qualitatively speaking only one degree of freedom remains after the decision to stay closer to the blue/bottom boundary. Making a choice for N is not as straightforward, as there are a few consequences. In fact, a lower N, i.e. a lower reference frequency f_0:

- increases the part-to-part spread on the nominal f_0, due to typically narrower springs.

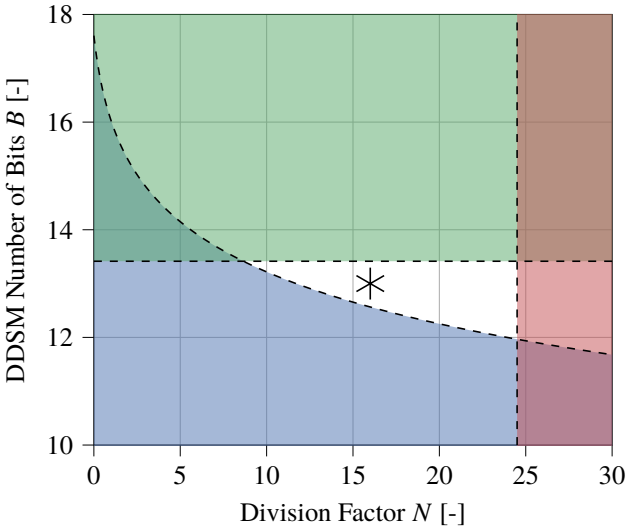


Figure 3.3: *Design space representation of the requirements expressed by* (3.12) *(maximum N, red area),* (3.15) *(minimum B, blue) and* (3.18) *(maximum B, green). The resulting white region is the set of allowed pairs (N, B). The asterisk marks the final choice made.*

- increases the effect of electrostatic softening, due to a lower f_0 hence a typically lower mechanical stiffness.
- decreases the consumption of the sustaining oscillator, due to a lower f_0.
- increases the short-term output frequency variation (i.e. jitter), due to a larger Δf between periods resulting from different division factor.

All in all, it was chosen a division factor $N = 16$ so that the target frequency is set at $f_0 = f_{rtc} \cdot 16.5 = 540\,\text{kHz}$. By the way, frequency divisions by a power of two are very convenient to implement with digital circuitry. As a consequence, the number of bits is set to $B = 13$ and, recalling (3.16), one obtains the required $T_{conv} = 250\,\text{ms}$

3.3 Requirements for Temperature Sensor

The temperature sensor was not designed during the course of this project. Nonetheless, to prove the feasibility of the concept proposed in this Thesis, the system requirement of this block were studied. The discussion that follows is aimed at proving the feasibility of the temperature sensor that would

be required, by finding the required number of bits B_T and the maximum tolerable offset T_{os}, inaccuracy T_{ia} and RMS noise σ_T and comparing these parameters with the state of the art.

It is important to clarify that it is assumed that a calibration on each fabricated RTC is available only at one temperature point. During this procedure, the RTC is thermalized at a nominal temperature T_R affected by an error T_ϵ, due to environmental condition or slight inaccuracy of the climatic chamber used. In this way, at the end of the calibration routine, each RTC will be associated with its *measured* temperature T'_R, different for all devices and equal to:

$$T'_R = T_R + T_\epsilon + T_{os} \tag{3.20}$$

and also with the corresponding measured reference frequency f'_0, different from the actual f_0 due to the temperature error:

$$f'_0 = f_0(T_R + T_\epsilon) = f_0 \cdot (1 + \alpha T_\epsilon + \beta T_\epsilon^2) \tag{3.21}$$

On one hand, the MEMS frequency drifts according to the actual temperature T :

$$f_0(T) = f_0 \left(1 + \alpha \cdot \Delta T + \beta \cdot \Delta T^2\right)$$

whereas, on the other hand, the model that is used to compute CW uses the measured frequency, the measured temperature variation $\Delta T_m = T + T_{os} + T_{ia} + \sigma_T - T'_R$ and the estimated $\overline{\alpha}$ and $\overline{\beta}$ so that it estimates the MEMS frequency as:

$$\widetilde{f_0}(T) = f'_0 \left(1 + \overline{\alpha} \cdot \Delta T_m + \overline{\beta} \cdot \Delta T_m^2\right)$$

One can notice that, recalling the definition of T'_R given by (3.20), the measured temperature displacement from the reference temperature is not dependent on T_{os}

$$\Delta T_m = T + T_{os} + T_{ia} + \sigma_T - (T_R + T_\epsilon + T_{os}) = T + T_{ia} + \sigma_T - T_\epsilon \tag{3.22}$$

To find the residual frequency drift at the output as a function of the temperature sensor parameters, one can begin by describing the frequency drift at the output as:

$$\frac{f_{out}(T)}{f_{rtc}} = \frac{f_0}{f_{rtc}} \cdot \frac{1 + \alpha \cdot \Delta T + \beta \cdot \Delta T^2}{N_{frac}(T)} \tag{3.23}$$

Expanding the $N_{frac}(T)$ expression from (3.5), and remembering that CW is computed by the TCM using the measured temperature variation ΔT_m,

one obtains that the frequency at the output can be expressed as follows:

$$\frac{f_{out}(T)}{f_{rtc}} = \frac{f_0}{f_{rtc}} \cdot \frac{1 + \alpha \cdot \Delta T + \beta \cdot \Delta T^2}{\frac{f_0'}{f_{rtc}} \cdot \left(1 + \overline{\alpha} \cdot \Delta T_m + \overline{\beta} \cdot \Delta T_m^2\right)} = \\ = \frac{f_0}{f_0'} \cdot \frac{1 + \alpha \cdot \Delta T + \beta \cdot \Delta T^2}{1 + \overline{\alpha} \cdot \Delta T_m + \overline{\beta} \cdot \Delta T_m^2} \tag{3.24}$$

To analyze the effect of an imperfect temperature measurement, one can linearize (3.24) while assuming a perfect knowledge of the α, β coefficients, to highlight the effect of the sensor non-idealities only, thus obtaining:

$$\frac{f_{out}(T)}{f_{rtc}} \approx \frac{f_0}{f_0'} \cdot \left(1 + \alpha \cdot \Delta T + \beta \cdot \Delta T^2\right) \cdot \left(1 - \alpha \cdot \Delta T_m - \beta \cdot \Delta T_m^2\right) \tag{3.25}$$

If one performs the multiplication, expanding f_0' according to (3.21), and neglects the 2$^{\text{nd}}$-order terms, the following simplified equation is obtained:

$$\frac{f_{out}(T)}{f_{rtc}} \approx 1 + \alpha \cdot (\Delta T - \Delta T_m - T_\epsilon) + \beta \cdot \left(\Delta T^2 - \Delta T_m^2 - T_\epsilon^2\right) \tag{3.26}$$

which implicitly contains within ΔT_m all the imperfections related to the temperature sensor (T_{ia} and σ_T) apart from the offset T_{os}, for which a single calibration is enough to cancel its effect. For example, to study the impact of T_ϵ, one can differentiate (3.26) with respect to T_ϵ:

$$\delta\left(\frac{f_{out}(T)}{f_{rtc}}\right) \approx [\alpha - \alpha + 2\beta \cdot T_\epsilon - 2\beta \cdot \Delta T_m] T_\epsilon = \\ = -2\beta \cdot (\Delta T_m - T_\epsilon) \cdot T_\epsilon \approx \\ \approx -2\beta \cdot \Delta T \cdot T_\epsilon \tag{3.27}$$

where the following approximations were used: $T_\epsilon \ll \Delta T_m$ and $\Delta T_m \approx \Delta T$.

From this, it is possible to infer that if $f_0(T)$ were a linear function of temperature, this system could tolerate a very large error T_ϵ from the calibration machinery. In fact, a 1-point calibration is enough to completely cancel the negative impact of T_ϵ. Things change when there is a higher-order dependence, such as in the present case: the term β introduces a temperature-dependent frequency error that is proportional to both ΔT and to T_ϵ. By imposing that the frequency error is much smaller that the $\Delta f / f$

specification, one can obtain the requirement on T_ϵ:

$$d\left(\frac{f_{out}}{f_{rtc}}\right) = -2\beta \cdot \Delta T \cdot T_\epsilon \ll \frac{\Delta f}{f} \Rightarrow$$
$$\Rightarrow T_\epsilon \ll \frac{\Delta f}{f} \cdot \frac{1}{2\beta \cdot \Delta T_{max}} = 3.1\,\mathrm{K} \tag{3.28}$$

which is far from a prohibitive condition to reach.

As for the resolution of the temperature sensor, with a similar procedure it is possible to extract that the temperature resolution σ_T must be:

$$\delta\left(\frac{f_{out}}{f_{rtc}}\right) = -\left(\alpha + 2\beta\Delta T\right)\sigma_T \ll \frac{\Delta f}{f} \Rightarrow$$
$$\Rightarrow \sigma_T \ll \frac{\Delta f}{f} \cdot \frac{1}{\alpha} = 0.33\,\mathrm{K} \tag{3.29}$$

where the 2$^{\text{nd}}$-order term β was neglected as it has a negligible impact in practice. The sensor part-to-part inaccuracy T_{IA} should also satisfy the same condition as σ_T, as they share the same dependence:

$$T_{IA} \ll \frac{\Delta f}{f} \cdot \frac{1}{\alpha} = 0.33\,\mathrm{K} \tag{3.30}$$

Given the resolution, one can compute the required number of bits of the sensor B_T as:

$$B_T = \log_2\left(\frac{T_{max} - T_{min}}{\sigma_T}\right) = \log_2\left(\frac{85\,^\circ\mathrm{C} + 40\,^\circ\mathrm{C}}{0.33\,^\circ\mathrm{C}}\right) = 8.5 \tag{3.31}$$

B_T is less than the required B, i.e. the number of bits of the DDSM. This is due to the fact that the spread of f_0 due to process imperfection ($\approx \pm 2\,\% = 20\,000$ ppm) is larger, by a factor of 10, than the spread due to temperature variation ($\approx \alpha\Delta T_{max} = \pm 1875$ ppm). Therefore, the 3 MSBs of CW are only determined by the process offset, whereas the remaining (at least) 9 LSBs are determined by a fixed calibration word summed to a term that compensates the temperature drift.

A temperature sensor with the performances described, with just one calibration point, with compatible area occupation, and consumption has been described in several works, such as [17], [26], [27][3].

[3]The interested reader can find several papers showing state-of-the-art temperature sensors listed in Prof. Makinwa's survey [25]

3.4 Jitter Suppression

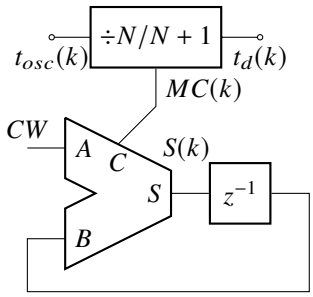
The authors of [17], that is the most comparable paper to the present work in terms of application and requirements, has achieved the required jitter suppression through the use of a Phase-Locked Loop (PLL). The PLL is used as a Low-Pass Filter (LPF) in the phase domain: a Voltage-Controlled Oscillator (VCO) is locked to the frequency – and phase – of the reference coming from the divided output of the fractional divider. The control system implemented by the PLL exploits a very low-bandwidth control signal and the phase/frequency inertia of the PLL to produce a cleaner periodic signal that shares the average frequency with the dirtier divided signal.

The PLL is very critical block: it requires a very careful design to stay within a consumption of few hundreds of nA, a fraction of the total 1 µA consumption for the whole system. Also, it requires relatively large passive components, since a loop bandwidth of ≈1 kHz is required to properly filter the noise due to the ΣΔ modulation. In this condition, the optimization of the jitter-vs-power trade-off for ring-based VCOs becomes difficult [28]. Advanced techniques, such as that described in [28], can be used to decrease the size of the passive components, but this increases even more the complexity of the design.

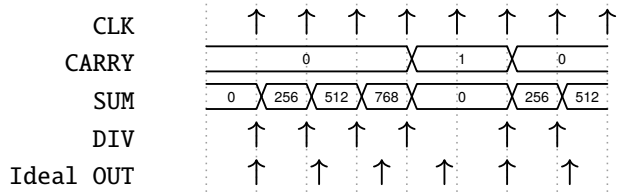
Therefore, in this work a different approach is proposed, which consists in a *compensation* of the jitter, as opposed to its *filtering*, in a fashion similar to what has been done in other fields of application such as that of frequency synthesizers [29], [30]. The implementation of this concept also led to the filing of a patent [4]. A fair comparison between the two approaches to jitter suppression requires a more in-depth description of the physical implementation of the two solutions. As a consequence, the rest of this Chapter is dedicated to the description of the proposed approach, whereas a theoretical comparison from the power-noise trade-off will be carried out at the end of Chapter 4.

3.4.1 DDSM Jitter: a Deterministic Process

The use of a PLL, that acts as a low-pass filter, inherently treats the jitter introduced by the DDSM operation as noise, i.e. a stochastic process, whereas actually it is not. This can be shown with the use of a simple example that refers to the use of a simple architecture for a Digital ΔΣ Modulator, shown in Figure 3.4a, that consists of a digital accumulator. In this architecture, the *CW* input is the only input of the accumulator, and the Modulus Control output is taken from the carry out (or overflow) bit of the block. The *CW* value gets accumulated in the registers that store the



(a) *Block scheme of the fractional divider.*



(b) *Timing diagram of the main signals involved in the operation of the DDSM.*

Figure 3.4: *The Digital ΔΣ Modulator is implemented as a digital accumulator with the carry-out bit as output, which is equivalent to a 1^{st}-order ΔΣ modulator. One can note that the time difference between the time edges of DIV and the ideal output is proportional to the value of SUM.*

sum value $S(k)$. The larger CW, the more often an overflow happens and the more often a division by $N + 1$, rather than N, happens. For the sake of clarity, a brief example with all the relevant digital signals is shown in Figure 3.4b.

A formal demonstration of the equivalence of the accumulator with a Digital $\Delta\Sigma$ Modulator can be found in [31, pp. 67-68]. With the same approach, it is possible to find a simple equation that describes the temporal position of the edges of the output clock. Let us define $t_d(k)$ as the discrete-time sequence of the temporal position of the sensitive edges of the divided clock. In an ideal case this sequence would be described by $t_d(k) = k \cdot T_{rtc}$. In the general case, this sequence assumes the following form:

$$t_d(k) = T_0 \left[k \left(N + \frac{CW}{2^B} \right) - \frac{S(k)}{2^B} \right] \tag{3.32}$$

The first term shows the average frequency compensation, dependent on CW, whereas the second term, dependent on $S(k)$, is the disturbance that affects the output phase. Since this disturbance depends on $S(k)$, which is a known digital value, stored in the registers of the accumulator, one can compensate this error by using a Digital-to-Analog Converter (DAC) or, better saying, a Digital-to-Time Converter (DTC). The quantization error can be completely canceled if the DTC block introduces a delay $\tau(k)$ equal to:

$$\tau(k) = \frac{S(k)}{2^B} \cdot T_0 \tag{3.33}$$

The application of this delay formally requires a time resolution of the DTC of $\frac{T_0}{2^B} = \frac{1}{540\,\text{kHz} \cdot 2^{13}} = 226\,\text{ps}$. This is very impractical, but one can decide to truncate the compensating delay to a more reasonable number of bits.

3.4.2 Jitter Evaluation

To correctly size the compensation block, an evaluation of the RMS period jitter introduced by the DDSM is necessary, to then compare this value with the $35\,\text{ns}_{\text{rms}}$ specification. From (3.32), the sequence $T_j(k)$ of the jitter affecting the periods corresponding to the divided clock, whose rising edges are placed at $t_d(k)$, results:

$$T_j(k) = t_d(k) - t_d(k-1) = \frac{T_0}{2^B} \left[S(k) - S(k-1) \right] \tag{3.34}$$

The fact that CW is constant, which is true between two samplings of the system temperature, entails that $S(k) - S(k-1)$ can assume only two values:

either CW or $2^B - CW$, according to if an overflow has or has not happened respectively. Therefore, over a complete limit cycle of – at most – 2^B periods of this system, $T_j(k)$ assumes the following values for the stated frequencies:

$$T_j = \begin{cases} \dfrac{T_{hf}}{2^B} \cdot CW & \text{for } 2^B - CW \text{ times} \\ \dfrac{T_{hf}}{2^B}\left(2^B - CW\right) & \text{for } CW \text{ times} \end{cases} \tag{3.35}$$

which corresponds to the probability density function for the DDSM jitter. From (3.35), it is finally possible to compute the RMS jitter σ_j in the case where no jitter suppression is employed:

$$\begin{aligned} \sigma_j^2 &= \overline{T_j^2} = \\ &= \left(\frac{T_0}{2^B}\right)^2 \frac{CW^2 \cdot (2^B - CW) + (2^B - CW)^2 \cdot CW}{2^B} = \\ &= \left(\frac{T_0}{2^B}\right)^2 \left(2^B \cdot CW - CW^2\right) \end{aligned} \tag{3.36}$$

Therefore, σ_j is a function of CW only, with a worst-case equal to $T_0/2 = 925\,\mathrm{ns_{rms}}$ when $CW = 2^{B-1}$.

The DTC can fully compensate this jitter but, for practical reasons such as power consumption, matching of components. . . , let us suppose that only the first B_c bits of $S(k)$ are used for the compensation and define $b = B - B_c$. Then, let us define the signals $\widetilde{S(k)} = S_{b-1:0}(k)$[4] and $\widetilde{CW} = CW_{b-1:0}(k)$, the values of the b LSBs of $S(k)$ and CW. When the DTC is employed, it is possible to describe the output jitter statistics in the same way, by using $\widetilde{S(k)}$ and $\widetilde{CW}$ in place of $S(k)$ and CW:

$$T_j = \begin{cases} \dfrac{T_{hf}}{2^B} \cdot \widetilde{CW} & \text{for } 2^b - \widetilde{CW} \text{ times} \\ \dfrac{T_{hf}}{2^B}\left(2^b - \widetilde{CW}\right) & \text{for } \widetilde{CW} \text{ times} \end{cases} \tag{3.37}$$

from which the RMS output jitter assumes a very similar expression to (3.36):

$$\sigma_j^2 = \left(\frac{T_0}{2^B}\right)^2 \left(2^b \cdot \widetilde{CW} - \widetilde{CW}^2\right) \tag{3.38}$$

[4]The syntax $A_{m-1:0}$ indicates the value assumed by the digital word formed by the last m bits of signal A.

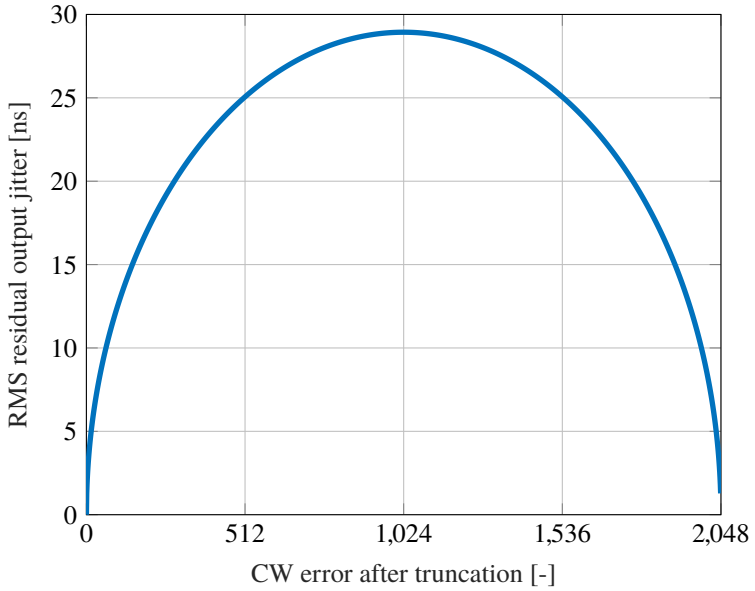


Figure 3.5: *Output RMS jitter as a function of the digital signal* $\widetilde{CW}$.

The worst-case output jitter σ_j^{wc} is obtained when $\widetilde{CW} = 2^{b-1}$ and is equal to:

$$\sigma_j^{wc} = \frac{T_0}{2^{B-b+1}} = \frac{T_0}{2^{B_c+1}} \tag{3.39}$$

Finally, from (3.39) it is possible to compute the number of bits B_c that the DTC is required to compensate to keep the jitter under specification imposing $\sigma_j^{wc} < 35$ ns:

$$B_c > \log_2\left(\frac{T_0}{2\sigma_j^{wc}}\right) = 4.7 \rightarrow 5 \tag{3.40}$$

A plot of the expected $\sigma_j(CW)$ is reported in Figure 3.5.

CHAPTER 4

Integrated Implementation

This Chapter discusses the aspects related to the physical on-silicon implementation of the RTC. The possible architectures and the practical details will be analyzed for each of the designed blocks, i.e. the oscillator, the fractional divider and the Digital-to-Time Converter (DTC). Finally, the simulated consumption and performances will be outlined. Parts of this Chapter have been published in [3] and the implementation of the DTC is the object of an application to an Italian patent [4], then extended to the USA.

4.1 Introduction

The system described in the previous chapters has been integrated on silicon, with the exception of the temperature sensor and the TCM. The Integrated Circuit (IC) was designed in the STM HCMOS9LP process, a 130 nm CMOS technological node by STMicroelectronics targeted for low-power applications and commonly used for many commercial products such as inertial sensors. This process features two types of transistors:

- thin-oxide transistors with minimum channel length of 130 nm and

rated for a maximum gate voltage of 1.2 V, referred to as GO1 transistors. . .

- . . . and thick-oxide transistors with minimum channel length of 0.5 µm and rated for a maximum gate voltage of 4.6 V, referred to as GO2 transistors.

The threshold voltages of all the transistors are nominally between 0.55 V and 0.6 V.

Metal-Insulator-Metal (MIM) capacitors are available, with two density options: 5 fF µm^{-2} and 2 fF µm^{-2}. The lower density option was chosen, due to the worry that the higher-density MIM capacitor could have shown a much bigger leakage. This leakage is usually negligible for common applications but could have been significant with a tight 1 µA current budget.

The power budget was split roughly equally between the three functions implemented within the system: the reference oscillator, the fractional $\Delta\Sigma$ modulated divider together with the DTC and the temperature sensor. This choice is a good rule of thumb and was found to be a good reference following what had been done in the literature and was retained as a guideline while practically implementing the circuit. Therefore, the reference for each of these block will be a third of the total 1 µA budget, i.e. a cautious 300 nA each.

The supply voltage was agreed with the industrial partner and is set at 1.2 V. As will be explained later in the Chapter, the digital blocks are operated from a 0.8 V supply, so that in the following these voltages will be referred to as the analog supply $V_{DDA} = 1.2\,\text{V}$ and the digital supply $V_{DDD} = 0.8\,\text{V}$.

4.2 Oscillator

The role of an oscillator is to make a resonant element (MEMS, quartz, LC tank) oscillate at its resonance frequency by injecting, at each oscillation cycle, the energy that this element dissipates due to its non-idealities. In this way, the oscillator circuit draws power from the supply to sustain the oscillation, whose frequency is set by the sharp frequency selectivity of the resonator at the resonant frequency. Then, the final output of the oscillator must assume the form of a square-wave voltage signal, so that it can be used as a clock by the rest of the system.

The oscillator design described in the following targets a MEMS resonator that is different from the one described in Chapter 2. It was designed by Gabriele Gattere from STMicroelectronics and shares the same design

as the previous version. The only difference is that the gaps are set by design at 400 nm, with an expected loss of critical dimension (CDLoss) of 300 nm, which yields an expected average gap of 700 nm. This choice was made possible by the use of a different etching machine and enables a more effective actuation. Since the motional resistance is proportional to the gap to the fourth power, this lowers the motional resistance down to $\approx$100 kΩ, which is very beneficial for the consumption of the oscillator, as will be shown later.

In the following sections the choice of the resonator topology and its circuital design will be presented.

4.2.1 Oscillator Topologies

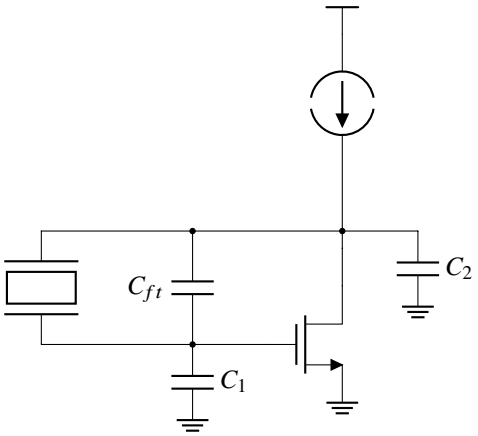
The two oscillator architectures analyzed for this project are the Pierce oscillator and a similar topology proposed by Ruffieux in [32], [33], that will be referred to as "cross-coupled oscillator".

Pierce Oscillator

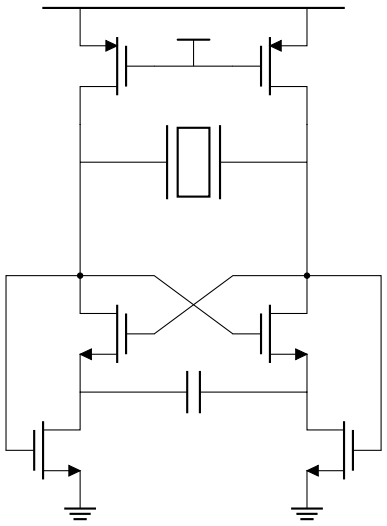
The most common oscillator used for MEMS (or quartz) resonators is the Pierce oscillator, whose most basic schematic is shown in Figure 4.1a. This popularity does not come without reasons. A clear cause for its widespread use is the fact that the Pierce oscillator is one of the single-transistor oscillator topologies. This means that the basic Pierce circuit can work by biasing only one branch: all the current available is converted in transconductance of the active element of the circuit. This effectively maximizes the efficiency in terms of transconductance per total current consumed, a fact very appealing when dealing with very strict power budgets. Additionally, it is the only one out of the three single-transistor oscillators – the Colpitts, the Clapp and the Pierce oscillators – that does not need a bias current through its inductor. This would not be compatible with the fact that capacitive MEMS resonators have an infinite impedance at DC.

Moreover, when working with very low currents, it becomes nearly impossible to synthesize low-impedance nodes: too much current would be required to create a low-impedance $\frac{1}{g_m}$ source node or to bias circuits with bandwidths so large to create virtual grounds. The Pierce oscillator overcomes this problem by not needing low-impedance nodes at all. In fact, it only uses a current generator with ideally infinite output impedance.

Despite being a circuit with a very low number of components, this circuit has been object of many studies since its first publication in 1923 [34]. The first person able to demonstrate virtually all the properties of this circuit has



(a) *Pierce oscillator basic schematic.*



(b) *Cross-coupled oscillator schematic.*

Figure 4.1: *Schematics of the considered oscillator topologies.*

been Vittoz in 1988 [22].

The theory developed by Vittoz explains that, in order for a Pierce oscillator to start up, a minimum transconductance for the active element is required, called critical transconductance and indicated as $G_{m,c}$. This parameter can be determined by imposing that the (negative) real part of the impedance synthesized by the oscillator circuit is larger than the series resistance of the resonator R_m, reaching the following expression:

$$G_{m,c} = R_m \omega_0^2 C_1 C_2 \tag{4.1}$$

which, considering a MOS transistor working in sub-threshold region for maximum g_m/I efficiency, provides the value for the minimum required current I_{crit} to bias the oscillator:

$$I_{crit} = G_{m,crit} \cdot nU_{th} = R_m \omega_0^2 C_1 C_2 nU_{th} \tag{4.2}$$

where n is the sub-threshold slope coefficient, assumed equal to 1.5, and U_{th} is the thermal voltage, equal to 25.8 mV at room temperature. The MEMS feedthrough capacitance C_{ft} has been considered much smaller than C_P.

Considering that $C_1, C_2 \approx C_P$, where $C_P \approx 0.9\,\text{pF}$ is the parallel composition of the parasitic capacitances of the I/O pads used to connect the IC and the MEMS, the active capacitance of the MEMS and the contributions due to the wire-bondings, the minimum theoretical current required for the Pierce architecture is:

$$I_{crit} = R_m \omega_0^2 C_P^2 nU_{th} = 36\,\text{nA} \tag{4.3}$$

with $R_m = 100\,\text{k}\Omega$, $\omega_0 = 2\pi \cdot 540\,\text{kHz}$.

Cross-Coupled Oscillator

A further interesting option is the cross-coupled oscillator proposed by Ruffieux *et al.* in [32], whose schematic is reported in Figure 4.1b. This circuit is equivalent to a differential Pierce oscillator. In fact, neglecting the two NMOS transistors at the bottom and the bridge capacitor, the schematic is that of a Pierce oscillator mirrored to create an NMOS differential pair connected in positive feedback, biased by the two PMOS current generators at the top. The two NMOS transistors at the bottom are then added to create a network that sets the common mode voltage at the MEMS terminals. The bridge capacitor is needed to create an AC short at the oscillation frequency and, at the same time, block the connection at DC, allowing a stable biasing of the circuit, which would otherwise be unstable locking the circuit in a state where no oscillation can arise.

In [32], the critical transconductance $G_{m,c}$ required to sustain an oscillation is given as:

$$G_{m,c} = 4\sqrt{\frac{C_m}{L_m}}\left(\frac{C_{ft}}{C_m}\right)^2 \frac{1}{Q}\left(1+\frac{C_P}{2C_{ft}}\right)^2 \tag{4.4}$$

where C_{ft} is the feedthrough capacitance of the MEMS resonator. Under the approximation $C_{ft} \ll C_P$, the $G_{m,c}$ of the cross-coupled oscillator is exactly the same given by (4.1) for the Pierce oscillator.

Since $G_{m,c}$ in this case refers to the transconductance of each one of the cross-coupled NMOS transistors, this topology requires twice as much current as the aforementioned Pierce topology:

$$I_{crit} = 2R_m\omega_0^2 C_P^2 n U_{th} = 72\,\text{nA} \tag{4.5}$$

Pierce vs Cross-Coupled Oscillators

As shown, the cross-coupled oscillator has, theoretically, a minimum consumption that is twice as that of the Pierce oscillator, a difference of the utmost importance in an ultra-low-power system such as this RTC. Nevertheless, both minimum consumptions are well within the guideline specification of 300 nA, so that neither of the two solutions can be discarded based solely on this aspect.

Another interesting aspect to compare the two topologies is the gain margin required for the oscillator startup. In fact, the consumption stated by (4.2) and (4.5) are valid only for the steady-state condition. At the startup, those conditions must be satisfied by a safety margin, called "gain margin", in order for the oscillation to reliably start-up.

One possibility, employed in several oscillator applications, such as [35], is to design the loop gain of the circuit reliably, i.e. for any process corner or environmental condition, larger than 1 in magnitude, usually by few dB. As a consequence, applying this strategy would mean that this oscillator would consume at least twice the achievable current consumption, which is sub-optimal.

A second possibility is the use of an Automatic Gain Control (AGC) circuit, an additional loop that sets the loop gain modulus of the primary oscillator circuit. Such loops have been proposed in the previously cited articles. Both these AGC circuits exploit the non-linearity of an MOS transistor to detect the sinusoidal voltage amplitude applied to the device. The sensed amplitude is zero when the circuit is switched on, but it increases exponentially in time. When this amplitude reaches a threshold, which is intrinsically built within the circuit, the bias current is reduced, so that the

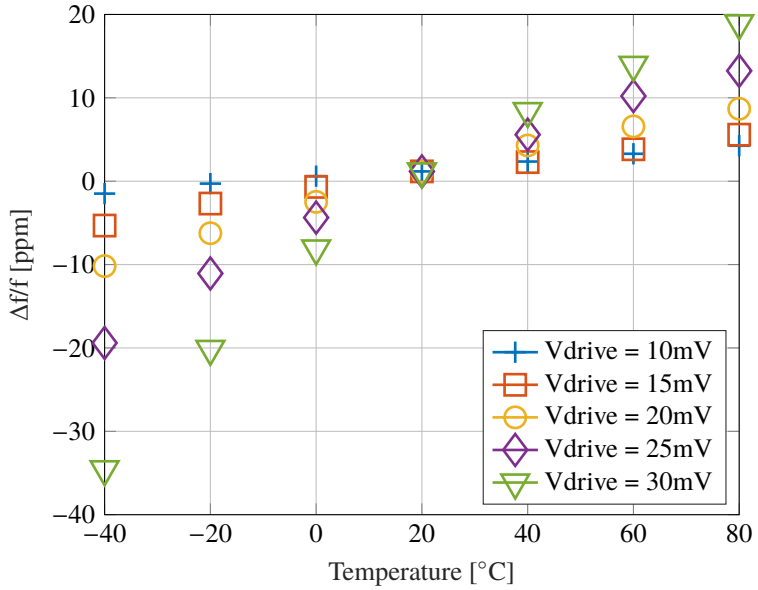


Figure 4.2: *$f_0(T)$ dependence of an ideal oscillator for different voltage drive levels. The expected drift in linear condition has been subtracted from the simulation shown in the plot, so that a linear MEMS resonator trace would appear as a horizontal line at* 0 ppm.

exact conditions expressed by (4.2) and (4.5) are exactly satisfied. In this condition, the gain margin must be adopted only at the startup, when any additional consumption is *de facto* irrelevant. The current consumed at steady-state, instead, can potentially reach the minimum theoretical value, summed to an overhead given by the consumption of the AGC itself.

The use of an AGC is also suggested by another key aspect: the $f_0(T)$ dependence of the MEMS resonator in non-linear regime. When the MEMS is driven with a large sinusoidal voltage, different non-linear phenomena can arise. In this case, the electrostatic field within the sensing/actuation parallel plate electrodes introduce a electrostatic stiffness k_{el} that, summed to the implemented mechanical stiffness k_m, changes the resonance frequency from $\sqrt{\frac{k_m}{m}}$ to $\sqrt{\frac{k_m+k_{el}}{m}}$, where m is the modal mass. This additional term is known and unavoidable, but when the actuation voltage becomes too large, the displacement of the device becomes a significant fraction of the parallel-plate gap and the $f_0(T)$ dependence changes. A simulation of this phenomenon was carried out using a VerilogA model of the specific MEMS, which encompasses the non-linearities of the device. The result is shown in Figure 4.2, from which the frequency deviation from the MEMS native $f_0(T)$

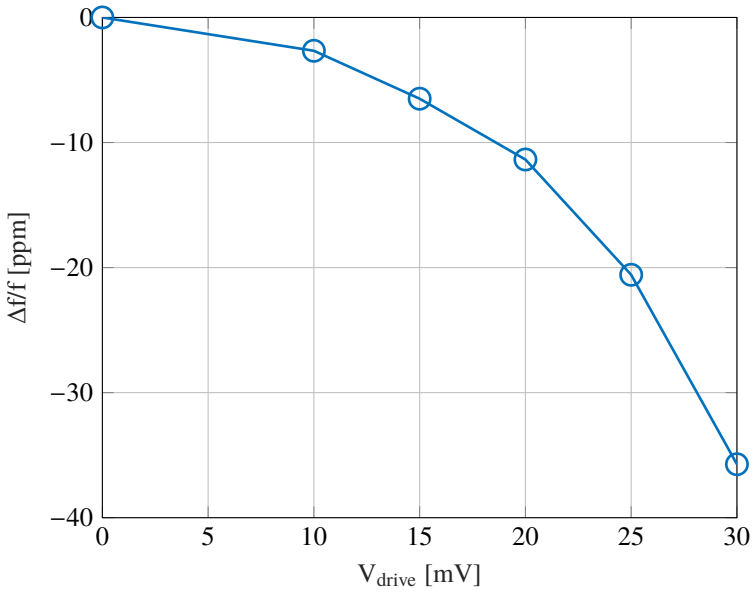


Figure 4.3: *$f_0(T)$ dependence, as per the simulation shown in Figure 4.2, at −40 °C for different voltage drive levels*

is shown. The maximum deviation is reached at −40 °C, that is the situation plotted in Figure 4.3. Here it is possible to see that if the MEMS is actuated with $V_{drive} = 30\,\text{mV}$, the sensitivity of the frequency error with respect to a variation in the voltage amplitude is $\approx -3\,\text{ppm/mV}$. A $V_{drive} \approx 10\,\text{mV}$ was chosen as a good compromise between feasibility and frequency stability.

Once the necessity of an AGC has been introduced, it is possible to analyze the differences between the Pierce and the cross-coupled oscillators in this regard. The AGC for the cross-coupled oscillator, illustrated in [32], adds one branch that must be biased with the same current of the other two "main" branches, so this AGC entails a 50 % increase in consumption. This limitation is not present in the AGC of the Pierce topology, since the current that flows in the AGC branches can be set at an arbitrary multiple of that of the core oscillator.

This fact, together with the fact that the immunity to disturbances and power supply noise granted by the differential structure of the cross-coupled oscillator is not a key factor for this application, the Pierce topology was chosen. Therefore, the design of the oscillator discussed in the following section will reference the schematic depicted in Figure 4.4.

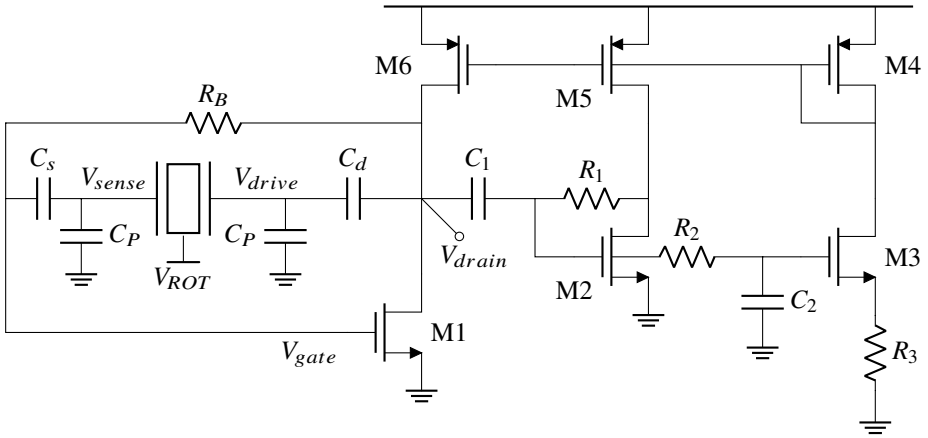


Figure 4.4: *Oscillator schematic: the branch made by M1/M6 constitutes the core of the oscillator whereas the other branches form the loop of the AGC.*

4.2.2 Core Oscillator

The goal of the core oscillator is to produce a negative resistance R_n that, for a Pierce oscillator, is expressed by:

$$R_n = -\frac{g_{m1}}{\omega_0^2 C_P^2} = -\frac{I_0}{nU_{th}} \cdot \frac{1}{\omega_0^2 C_P^2} \tag{4.6}$$

where it was assumed sub-threshold operation of M1 and I_0 indicates the steady-state bias current of transistors M1/M6. The discussion will now proceed to find the best strategy to reach the required oscillation condition at the lowest possible consumption.

AGC

To size the oscillator, it is possible to begin with the AGC loop formed by the branches that comprise M2, M5, M3, M4 (see Figure 4.4). This loop exploits the fact that a sinusoidal oscillation at V_{drain} is AC coupled to $V_{G,N2}$ by the high-pass filter created by R_1, C_1 ($\frac{1}{R_1 C_1} \ll \omega_0$). If this oscillation is large (at least comparable with nU_{th}), then:

1. M2 introduces harmonics in the current I_2 of sub-threshold operated M2, according to its exponential characteristic. Specifically, the DC current increases.

2. M3 retains its DC current, since its gate is driven by the low-pass filtered V_{drain} ($\frac{1}{R_2 C_2} \ll \omega_0$), and so do M4 and M5.

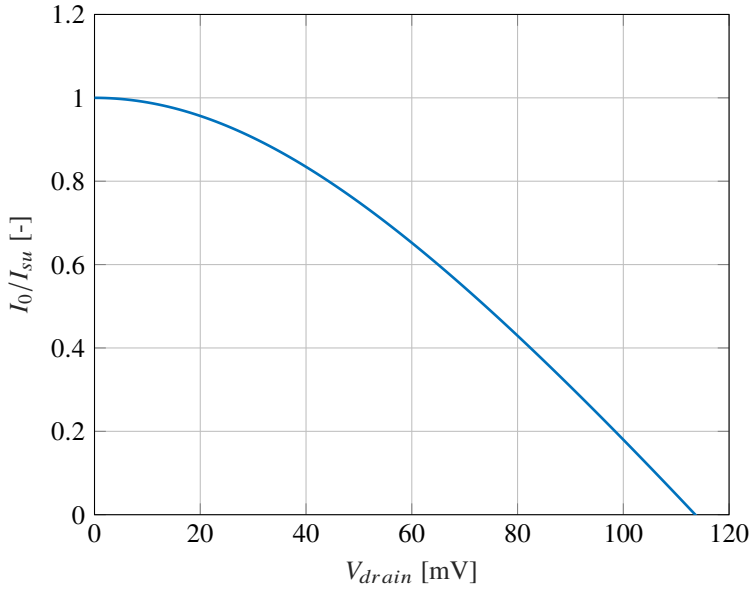


Figure 4.5: *AGC characteristic.*

3. The current balance at DC between I_2 and I_5 is not satisfied: since the Kirchhoff Current Law cannot be violated, the only possible bias point requires V_{G2} to drop statically. This reduces the DC current I_3, that is equal to I_4 and I_5.

Since this AGC is able to reliably control voltage amplitudes that are at least in the order of the thermal voltage, the controlled voltage V_{drain} appears attenuated at the MEMS terminal by a capacitive divider formed by C_d and C_P down to the chosen drive amplitude of 10 mV. These capacitances do not modify the behavior of the Pierce oscillator, because they are in series with ideally infinite impedances. The addition of the decoupling capacitances C_d and C_s also has the effect of maximizing the effective voltage between rotor and stators by biasing at 0 V the stators.

Considering equal lengths for corresponding transistors ($L_1 = L_2 = L_3$, $L_6 = L_5 = L_{P3}$), $W_5 = W_4$ and introducing $K_w = \frac{W_4}{W_5}$ and $K_I = \frac{W_6}{W_4}$, this loop sets the startup current I_{su} to the following value:

$$I_{su} = K_I \cdot \frac{U_{th}}{R_3} \ln(K_w) \tag{4.7}$$

and reduces it with increasing amplitude of the controlled signal V_{drain},

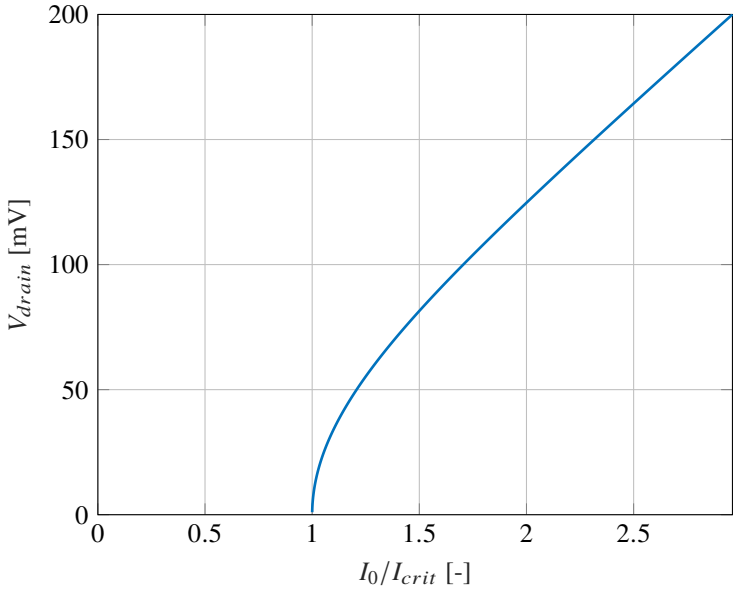


Figure 4.6: *Oscillation amplitude at net V_{drain} as a function of the bias current I_0.*

according to:

$$I_0 = I_{su}\left(1 - \frac{\ln\left[\mathcal{I}_0\left(\frac{V_{drain}}{nU_{th}}\right)\right]}{\ln(K_w)}\right) \tag{4.8}$$

which is represented in Figure 4.5.

The amplitude of the controlled voltage V_{drive} is set by the steady-state current I_0 according to the following equation:

$$\frac{I_{su}}{I_0} = \frac{V_{drain}}{nU_{th}} \cdot \frac{\mathcal{I}_0\left(\frac{V_{drain}}{nU_{th}}\right)}{2\mathcal{I}_1\left(\frac{V_{drain}}{nU_{th}}\right)} \tag{4.9}$$

where $\mathcal{I}_n(x)$ indicates the modified Bessel function of the 1st kind of order n with argument x. This expression is represented in Figure 4.6.

Combining (4.7), (4.8) and (4.9) is a good starting point to size the main parameters of the oscillator, then to be refined by simulation considering, for example, additional losses caused by non-idealities introduced by, e.g., the AGC and output buffer loading on the oscillator. In particular, the loading caused by R_1, C_1 and R_2, C_2 is critical in that the current that transconductor M1 would pump in the resonator to overcome its losses, can be partially (or substantially) diverted in the AGC, through the impedances that it places

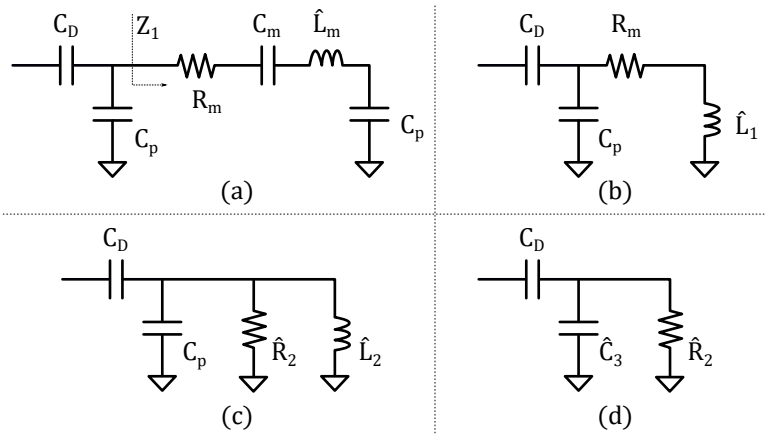


Figure 4.7: *Impedance transformations used to obtain the value for the capacitive partition of* (4.18).

in parallel to V_{drain}. Therefore capacitances C_1, C_2 were minimized and resistances R_1, R_2 were implemented as transistors in sub-threshold region.

In the end, it was decided to use $K_W = 6$, $R_3 = 430\,\text{k}\Omega$, $K_I = 5$ to have $I_{su} = 1\,\mu\text{A}$ and a steady state oscillation amplitude of $\approx$100 mV, an amplitude small enough so as not to worsen the steady state consumption but large enough to be squared easily.

Capacitive Divider

The capacitive attenuation caused by C_d, introduced to cancel the effect of the MEMS non-linearities, has a value that is different by the mere simple capacitive divider $\frac{C_d}{C_d+C_P}$.

To the aim of estimating the exact value of the partition V_{drive}/V_{drain} let us consider the portion of the oscillator circuit formed by C_d, the resonator and the two C_P, shown in Figure 4.7a. The impedance indicated as Z_1 is equal to:

$$Z_1(\omega_0) = R_m + j\omega_0 L_m + \frac{1}{j\omega_0 C_m} + \frac{1}{j\omega_0 C_P} \tag{4.10}$$

Since the exact oscillation frequency of this oscillator can be expressed as:

$$\omega_0 = \frac{1}{\sqrt{L_m \cdot \left(C_m \parallel \frac{C_P}{2}\right)}} \tag{4.11}$$

This means that the sum of the reactances of L_m, C_m and $C_P/2$ cancel each

other out at $\omega = \omega_0$. It is therefore possible to re-write (4.10) as:

$$Z_1(\omega_0) = R_m + \frac{j}{\omega C_P} = R_m + j\omega_0 \hat{L}_1 \tag{4.12}$$

where the equivalent inductance $\hat{L}_1$ was defined as:

$$\hat{L}_1 = \frac{1}{\omega_0^2 C_P} \tag{4.13}$$

To proceed, it is now convenient to resort to impedance transformations [36], a procedure that enables to transform series combinations of resistors and capacitors/inductors into parallel ones (and vice-versa) that finds equivalent circuits valid only at the exact frequency of operation. The original portion of the circuit is therefore equivalent – but only at frequency $\omega = \omega_0$ – to that sketched in Figure 4.7b. Let us define the quality factor $\hat{Q}$ of inductance $\hat{L}_1$ as:

$$\hat{Q} = \frac{\omega_0 \hat{L}_1}{R_m} = \frac{1}{\omega_0 R_m C_P} \tag{4.14}$$

This parameter is useful to proceed to the subsequent circuit transformation shown in Figure 4.7c, where parameters $\hat{R}_2$ and $\hat{L}_2$ are given by:

$$\hat{L}_2 = \hat{L}_1 \frac{1 + \hat{Q}^2}{\hat{Q}^2} \tag{4.15}$$

$$\hat{R}_2 = R_m \left(1 + \hat{Q}^2\right) \tag{4.16}$$

It is possible to notice that the parallel of C_P and $\hat{L}_2$ is equivalent to capacitance $\hat{C}_3$ (see Figure 4.7d):

$$\hat{C}_3 = \frac{C_P}{1 + \hat{Q}^2} \tag{4.17}$$

With the numbers used in this project, one can find $\hat{Q} \approx 3$ and capacitance $\hat{C}_3$ can be neglected with minimal error compared to the resistance in parallel $\hat{R}_2$. It is to be noted that, apart from this final approximation, the transformations done up to now are exact, even though valid only at $\omega = \omega_0$. Finally, approximating $1 + \hat{Q}^2 \approx \hat{Q}^2$, it is possible to find the desired partition between V_{drain} and V_{drive}:

$$\frac{V_{drive}}{V_{drain}} \approx \frac{1}{1 - j\omega_0 \frac{C_P}{C_d} C_P R_m} = \frac{1}{1 - j\frac{C_P}{\hat{Q} C_d}} \tag{4.18}$$

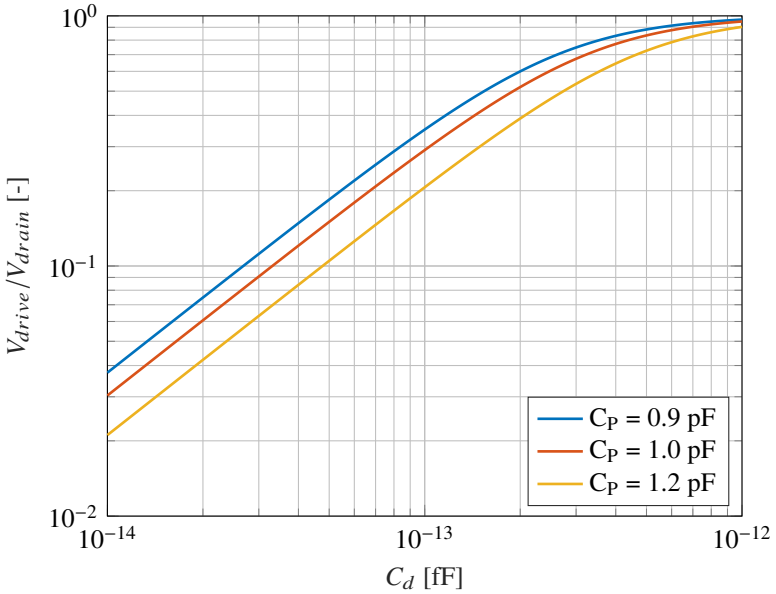


Figure 4.8: *Attenuation plotted as a function of C_d for different values of C_P.*

whose magnitude is:

$$\left|\frac{V_{drive}}{V_{drain}}\right|^2 = \frac{C_d^2}{C_d^2 + \frac{C_P^2}{\hat{Q}^2}} \tag{4.19}$$

which is plotted in Figure 4.8 for different values of C_P. In the end, considering a worst-case of $C_P = 0.9\,\text{pF}$, C_d was sized at 30 fF to reach an attenuation of a factor of 10.

Bias Network

A few auxiliary bias voltages need to be derived to bias the gates of the transistors that, with reference to Figure 4.4, implement resistor R_B and the two resistors in parallel to the two capacitors C_P, not shown in the figure for the sake of clarity. The bias network used, shown in Figure 4.9, works as a replica bias for transistor M7, so that the its resistance can remain relatively constant over the temperature range. Transistors M8-M9, instead, had to be much more resistive so that the same approach could not be used. Therefore, transistor M12 is introduced to produce a reduced gate voltage for M8-M9.

Overall, the loop gain of the oscillator was simulated over the whole

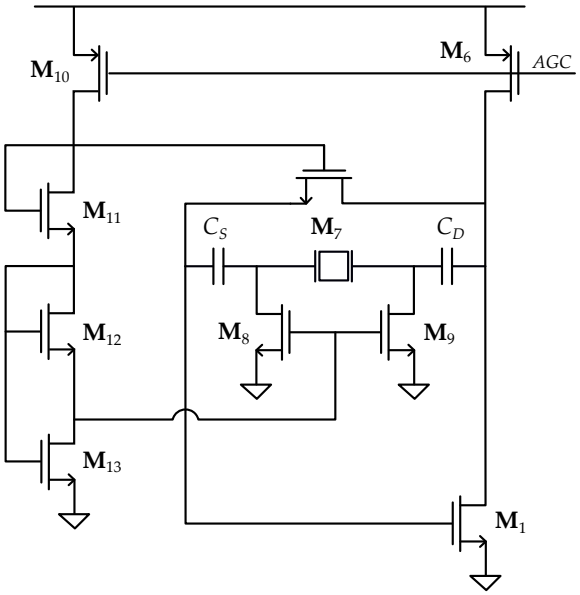


Figure 4.9: *Schematic of the Pierce core with the auxiliary bias branch. Transistor M7 works in the ohmic region and forms resistor R_B of Figure 4.4, that would otherwise occupy too much area. Transistors M8/M9 are also in their ohmic region and form two resistors of large value needed to correctly bias the MEMS terminals at* 0 V.

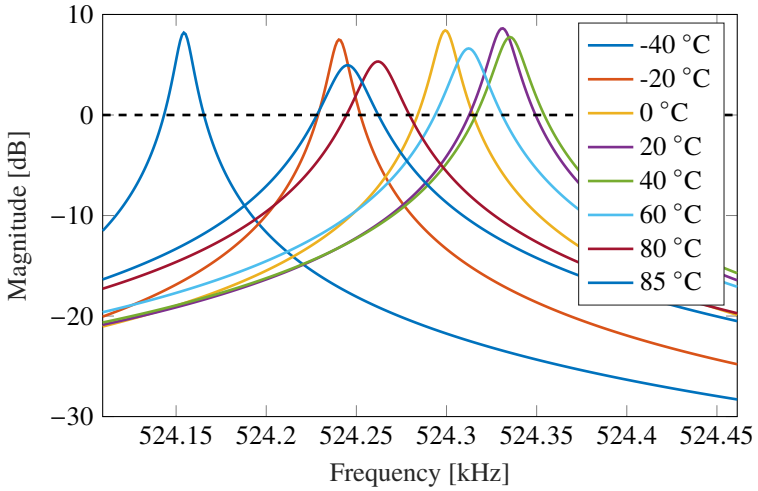
temperature range and across the main process corners. The result is shown in Figure 4.10 and shows a gain margin of at least 5 dB with a reliable and sharp crossing of the 0° axis over the whole −40 °C to 85 °C range.

Comparator

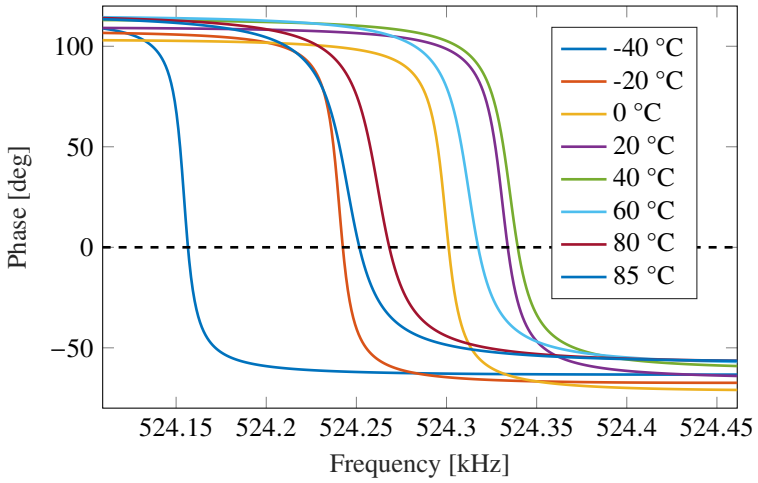
The comparator block must square the input V_{drain}, a 100 mV sinusoid around a DC value that can change according to the process/component statistical spread and the temperature. A simple inverter is not suitable: according to the particular input DC value and the threshold of the inverter, which is itself subject to spread, the small input signal may even never cross the comparator threshold.

Therefore, several alternative implementations were tested, which involved the linearization of the inverter characteristic by inducing a local feedback within the inverter block.

The most reliable of these approaches is represented in Figure 4.11. Here, transistors M21 and M22 have their V_{GS} = 1.2 V and act as degeneration resistors for transistors M19-M20, that for the inverter at the basis of the stage. A further linearization is achieved by using a resistor implemented



(a) *Magnitude of the loop gain: a* 5 dB *gain margin is obtained in the worst-case.*



(b) *Phase of the loop gain, which cross reliably the* 0° *axis over the whole temperature range.*

Figure 4.10: *Loop gain plotted for various temperatures. To optimize the visualization and the simulation time, the linear TCf was removed from the MEMS VerilogA model.*

Transistor	W/L [µm/µm]	Transistor	W/L [µm/µm]
M1	5/2.5	M12	25/0.5
M2	5/1	M13	2/10
M3	$6 \times (5/1)$	M14	8/1
M4	(3/5)	M15	8/1
M5	(3/5)	M16	1/15
M6	$5 \times (3/5)$	M17	15/1
M7	0.5/2	M18	8/1
M8	10/5	M19	1/1
M9	10/5	M20	1/1
M10	1/15	M21	1/15
M11	50/0.5	M22	1/15

Table 4.1: *Sizing of the transistors within the Pierce oscillator.*

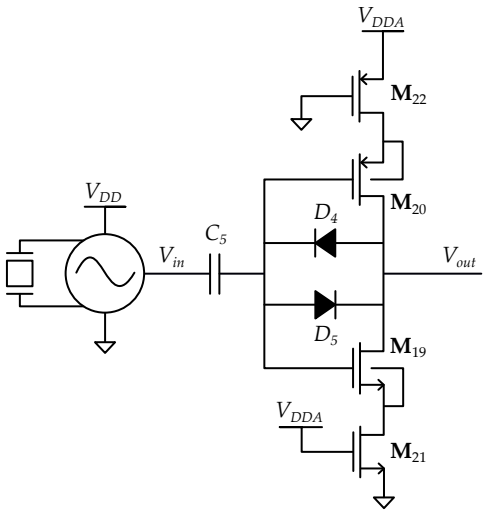


Figure 4.11: *Schematic of the implemented output buffer that must convert the sinusoidal signal from the oscillator to a squarewave output.*

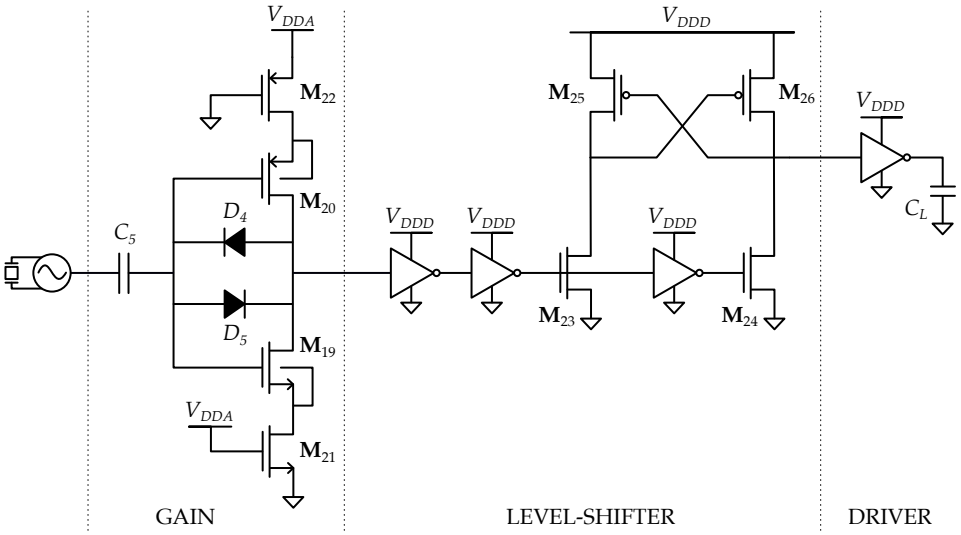


Figure 4.12: *Final schematic of the output buffer. The first stage constitutes a gain that outputs a rail-to-rail distorted sinusoid that is then squared by the 2 following inverters. A level-shifter then converts the logic level of the clock that is then used by the following blocks.*

with two diodes in anti-parallel configuration. This equivalent resistance is very large when no voltage is applied to the diodes, whereas it decreases with increasing voltage applied. The stage is AC-coupled to voltage V_{drain} through capacitor C_5 and, as a matter of fact, achieves a gain that is proportional to the ratio of C_5 and the equivalent capacitances of the diodes. $C_5 = 20\,\text{fF}$ was found to be a large enough capacitor to amplify V_{drain} to a level where a simple inverter could produce a sharp enough squarewave output.

Finally, a level-shifter was added to turn the 0 V to 1.2 V squarewave to a 0 V to 0.8 V output, the logic levels used by the subsequent digital block, reaching the final circuit represented in Figure 4.12.

4.2.3 Oscillator Performances

Finally, a snapshot of the most important waveforms within the oscillator at steady state is reported in Figure 4.13. The expected consumption, computed with Monte-Carlo simulations, is 345 nA on average, with a standard deviation of 49 nA. The detailed current consumptions subdivided between the three main blocks within the oscillator is reported in Table 4.3. The output jitter is expected to be $8\,\text{ns}_{\text{rms}}$ and an example of the oscillator startup is given in Figure 4.14.

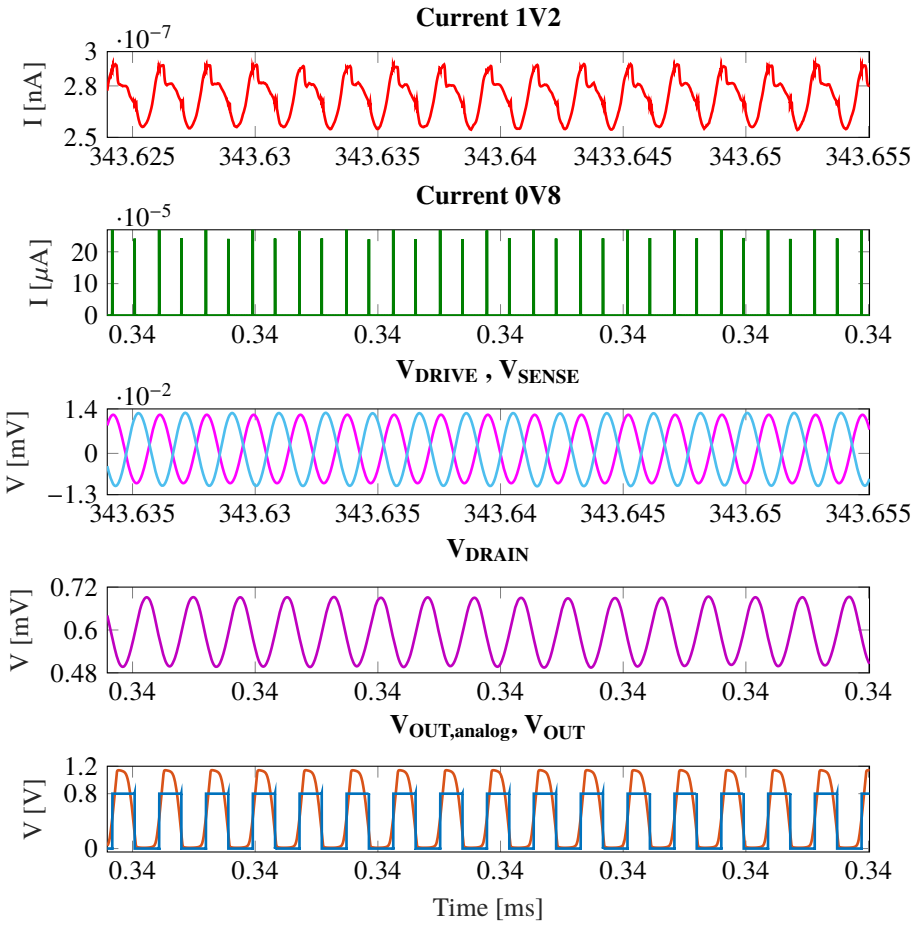


Figure 4.13: *Simulation of the main signals within the oscillator at steady state.*

Parameter	Value
K_w	6
K_I	5
R_3	430 kΩ
C_d	30 fF
C_s	500 fF
C_1	100 fF
C_2	6 pF
C_5	20 fF

Table 4.2: *Size of the components within the oscillator.*

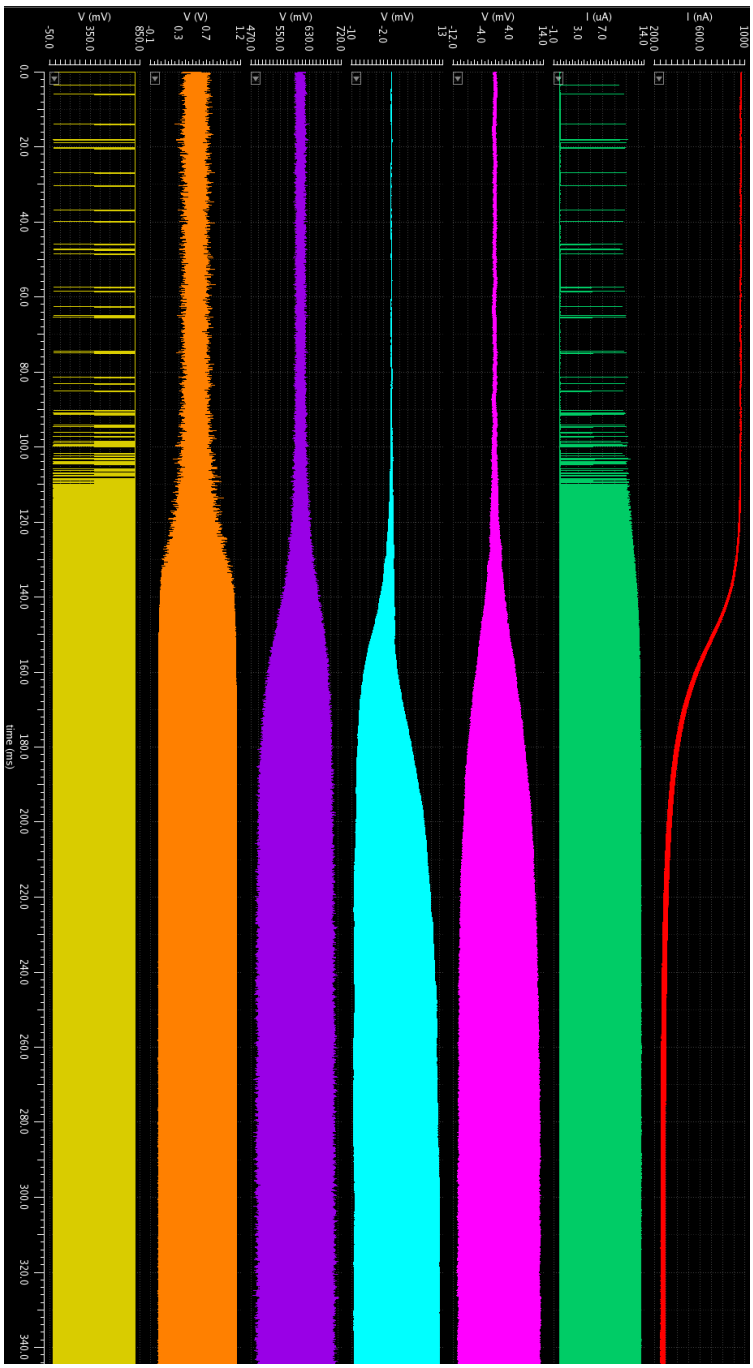


Figure 4.14: *Simulated oscillator startup with the most relevant waveforms.*

Consumption [nA]	min	avg	max	std dev
Oscillator	164.5	193.8	281.5	33.97
AGC	70.86	84.91	124.7	14.9
Comparator	63.76	67.81	72.38	1.84
TOT	301.8	345.5	477.2	49.37

Table 4.3: *Consumption of the oscillator subdivided in the constituent sub-blocks.*

4.3 Multi-Modulus Divider

The multi-modulus divider is a purely digital circuit that is required to divide its input clock frequency by either 16 or 17 according to the logic level of its Modulus Control (MC) input.

The current consumption of digital circuits I_{dig} is usually expressed as the sum of three contributions [37]: the dynamic consumption I_{dyn}, the leakage I_{leak} and the cross-conduction consumption I_{cc}. This can be expressed as:

$$I_{dig} = I_{dyn} + I_{leak} + I_{cc} = \sum_i \left(\alpha_{sw,i} C_i V_{DD} f_{ck}\right) + I_{leak} + I_{cc} \tag{4.20}$$

where C_i is the total capacitance hanging from net i and α_{sw} is the probability of having a low-to-high transition at net i, V_{DD} is the power supply voltage and f_{ck} is the clock frequency.

For this project, since the digital circuit to design is not critically complex and the clock frequency (540 kHz) is small with respect to typical clock frequencies allowed by the technology, it was decided to adopt a power supply for the digital portion of 0.8 V, that will be indicated as V_{DDD}, in contrast to the analog supply $V_{DDA} = 1.2$ V. This choice affects:

- I_{dyn}, that is decreased by 33 %
- I_{leak}, which is exponential with the supply voltage, is greatly reduced
- I_{cc}, that is substantially eliminated, since the thresholds of the used transistors are ≈0.6 V, a value very close to V_{DDD}.

At the same time, the digital circuitry remains fully functional for any process corner at $f_{ck} = 540$ kHz.

As an example to justify the importance of the care used to design the fractional divider and to provide the Reader with a feeling for the numbers involved, a capacitance of 10 fF, toggling between the low and high levels at 540 kHz, contributes to 6.5 nA of dynamic consumption.

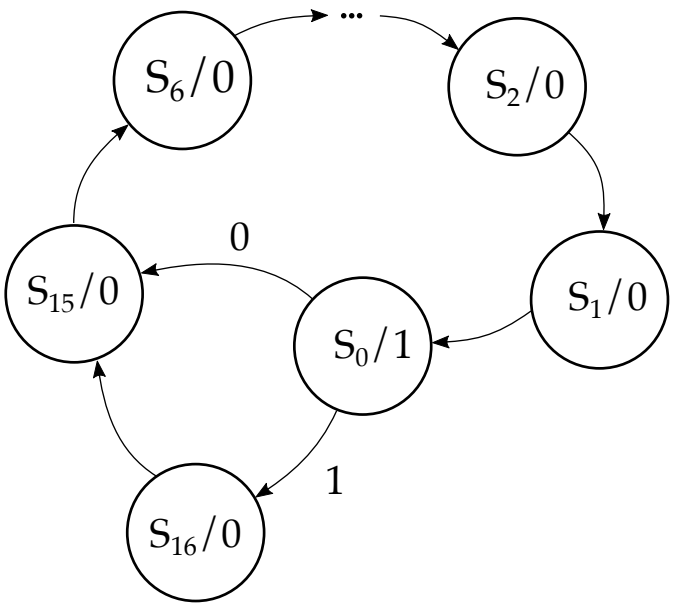


Figure 4.15: *Gray-encoded divider concept represented with its state diagram. When S_0 is reached, the counter is reset to S_{15} or S_{16} according to the value of MC.*

4.3.1 Custom Digital Design

The strategies used in the design of the divider are focused on (i) adopting digital logic families that require fewer transistors so as to decrease the switched capacitance C_i and (ii) reducing the switching activity α_{sw} of the most critical nets by acting at architectural level.

A first solution that was considered was a Finite-State Machine that synthesizes a Gray-encoded down-counter that employs 5 flip-flops (FF) clocked at f_{ck}, whose state diagram is illustrated in Figure 4.15. This solution was conceived as a way to reduce the switching activity of the nets of the FFs: since in the Gray encoding successive states differ by only one bit, this technique achieves the maximum possible reduction of the α_{sw} related to those nets.

A downside of this solution is that a relatively big combinational network must be introduced to let the system evolve in the correct future state given the present state. To minimize this problem, the combinational network was implemented with the *LEAn integration with Pass transistors* (LEAP) logic family [38]. This network is represented in Figure 4.16. The advantage of this logic family is that, given a logic function, it decreases the number of required transistors, hence the total capacitance and consumption, with

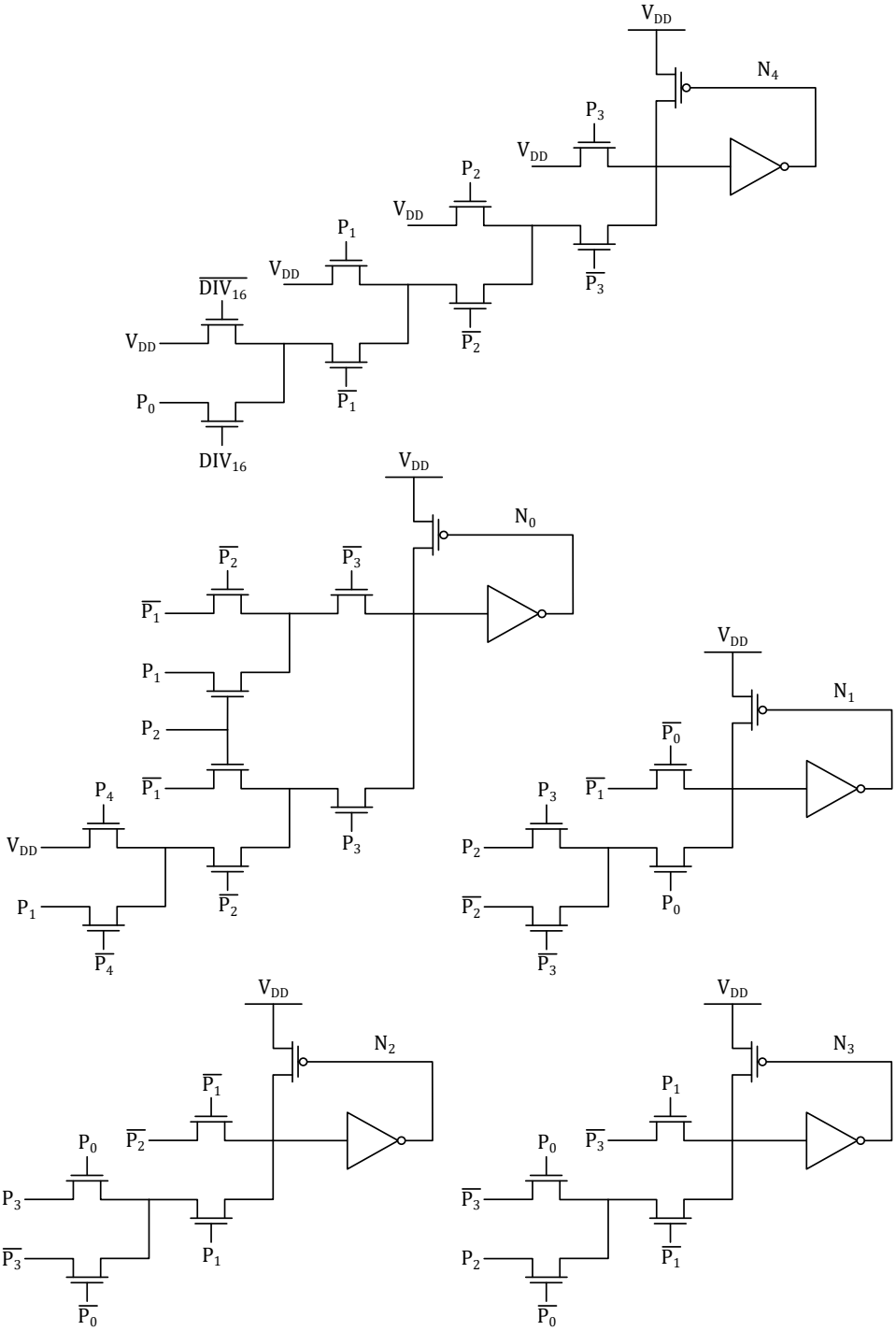
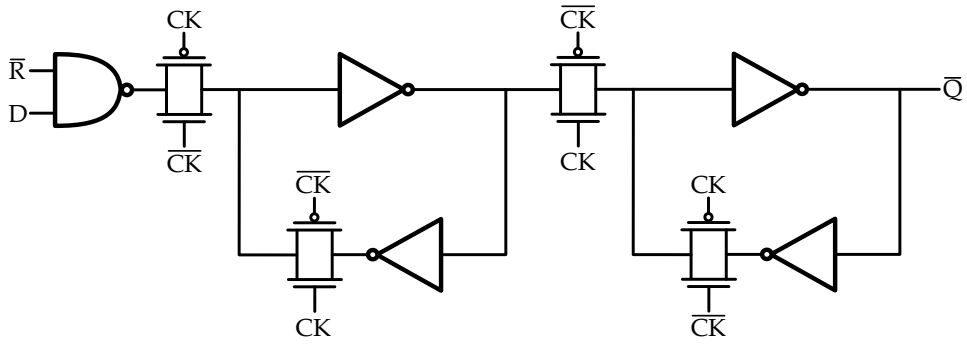
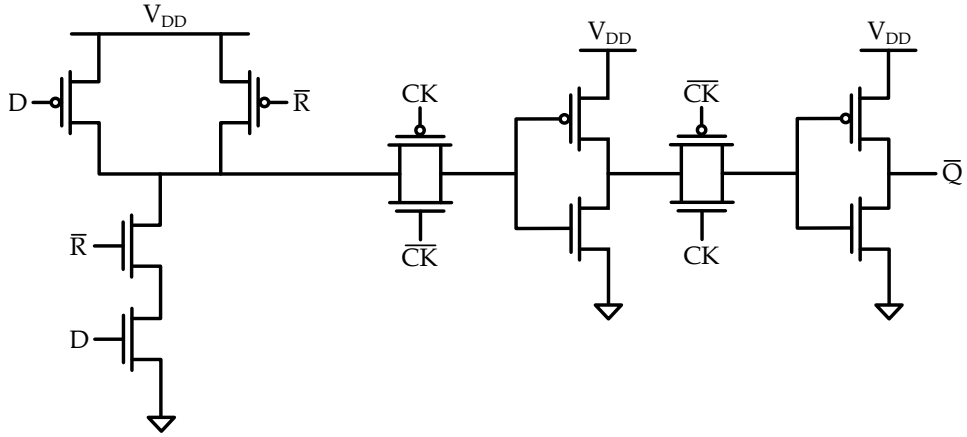


Figure 4.16: *Schematic of the combinational network implemented in LEAP logic of the Gray-encoded divider. Signals P_i indicate the present state bits, i.e. the outputs of the 5 FFs used, whereas signals N_i indicate the next state bits that are sampled by the FFs at the clock edge.*



(a) *Static FCCMOS flip-flop with synchronous reset.*



(b) *Dynamic flip-flop, adopting two buffered transmission gates.*

Figure 4.17: *Schematics of the flip-flops considered during the design phase.*

respect to more traditional logic families, such as the Fully-Complementary CMOS (FCCMOS). It achieves so by using a NMOS-only pass-transistor selection tree; this in turn requires a level restorer, made with a "weak" PMOS transistor and an inverter, to correctly drive the high output level up to the supply voltage.

The 5 FFs were implemented in a dynamic family, able to store a datum with a much lower number of transistors with respect to a static FCCMOS FFs (Figure 4.17a). The FF was implemented as a series of two transmission gates buffered by inverters, as shown in Figure 4.17b. A synchronous reset was introduced by means of a NAND gate that accepts as inputs the datum D and the active-low reset signal $\bar{R}$. All the transistors were designed with minimum length and width to reduce the total switched capacitance.

Even though this solution provided a significant reduction of the consumptions when using a 1.2 V supply, it was found that the lower 0.8 V

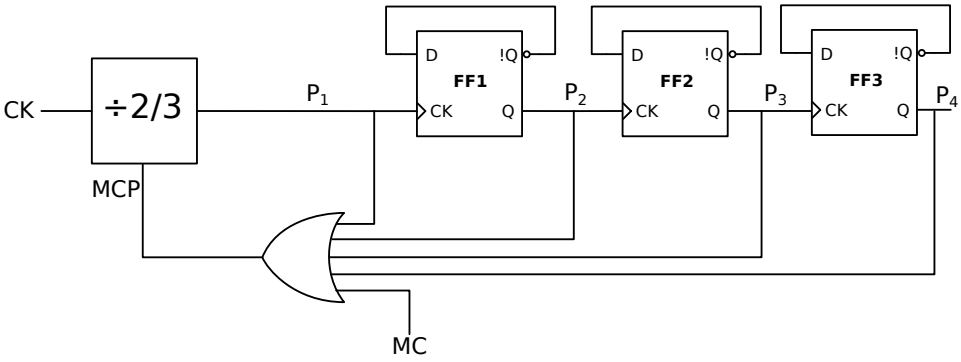


Figure 4.18: *Asynchronous divider with modulo-2/3 prescaler from [39]. MC sets the 16-17 division modulus through a 5-input OR gate that synthesize the Modulus Control of the Prescaler MCP. When a division by 17 is required, MCP dynamically changes its value so that the correct modulus is obtained.*

supply does not allow a reliable operation of LEAP circuits. In fact, the network incurs in logic errors on slow process corners and at low temperatures, i.e. when the transistors threshold are largest.

A second solution, more efficient at the lower 0.8 V supply, is based on the divider used in a 5 GHz frequency synthesizer by Pellerano *et al.* in [39], reported in Figure 4.18. This solution adopts a modulo-2/3 prescaler cascaded with 3 more FF in a divide-by-2 configuration, i.e. with their $\bar{Q}$ output shorted to the D input. This divider is an asynchronous counter, because each FF runs at half the clock frequency of the preceding one. This has beneficial effects on the consumption, since FFs have progressively lower α_{sw}. The combinational network, composed by a 5-input OR gate, imposes a modulus-3 division to the prescaler when its output MCP is low, i.e. when the Modulus Control signal MC is low and the 3 FFs all store a low datum. Therefore, if an overall division by 17 is required, the signal $f_{ck}/2$ gets stretched by one input period and all the remaining FF output are delayed by the same amount, thus stretching the output pulse from $f_{ck}/16$ to $f_{ck}/17$.

The circuit used for the prescaler is shown in Figure 4.19. In case the Modulus Control of the Prescaler bit (MCP) is '1', X is always high and flip-flop FF2 is in a divide-by-2 configuration, since the AND gate just buffers the negated output of FF2 to its D input. Therefore, if $MCP = 1$, this circuit is a modulo-2 frequency divider.

If $MCP = 0$, then this can be seen as a Finite-State Machine with 2 FFs and 4 possible states. The combinational network that drives the next state is built so that this circuit implements a down-counter that, once it

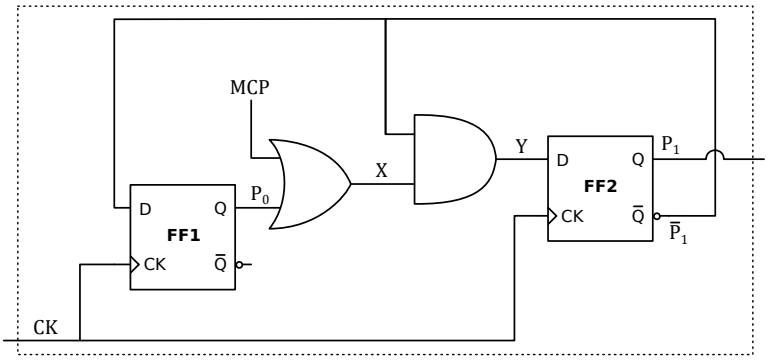


Figure 4.19: *Schematic of the prescaler circuit, taken from [39]. All the flip-flops in this component were dynamic, since they are all clocked at the maximum frequency of* 540 kHz *and, even at high temperatures, leakage does not erase the stored datum.*

reaches state $P_0/P_1 = 0/0$, it resets the bits to $P_0/P_1 = 1/0$, so that actually a division by 3, rather than 4, is achieved.

MCP is driven by a 5-input OR gate that takes as input the overall Modulus Control MC and the output values of the 3 cascaded FFs and the output P_1 of the prescaler.

This second option has the advantage of being asynchronous, because several flip-flops are clocked by sub-multiples of f_{ck}, and a relatively simple combinational network. Nonetheless, the dynamic FFs shown in Figure 4.17b with reduced dynamic consumption cannot be employed for the slower FFs of the chain. In fact, at the higher temperatures of the operating range, their datum is not refreshed often enough so that leakage does not erase their state. Because of this, adopting a conservative approach, the prescaler is designed with dynamic FFs to reduce the consumption of the fastest and most power-hungry devices of the circuit. The other FFs were designed in FCCMOS logic with reduced size of all the transistors.

The DDSM was implemented as an accumulator, equivalent to a 1^{st}-order modulator since an aggressive quantization noise shaping towards the Nyquist frequency is not needed. This circuit is clocked, on average, at 32.768 kHz, so that its consumption is not critical. Therefore, it was designed with standard cells (Full Adders and FFs).

The overall consumption of the fractional divider and the DDSM is 32 nA.

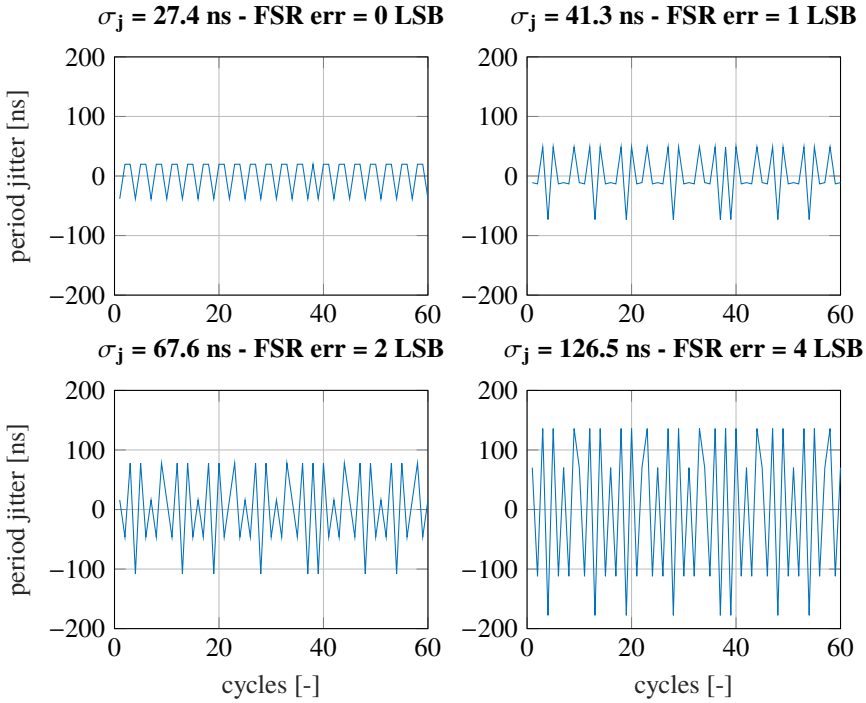


Figure 4.20: *Residual output jitter after a 5 bit compensation with an error on the FSR of the DTC. The RMS period jitter value and the error magnitude is indicated in the title of each subfigure.*

4.4 Jitter-Suppressing DTC

In Section 3.4 the deterministic nature of jitter in this system was introduced. This jitter can in principle be completely compensated by applying a delay τ to the output clock edges equal to:

$$\tau(k) = \frac{T_0}{2^B} \cdot S(k) \tag{4.21}$$

It was also determined that $S(k)$ can be truncated to the first 5 bits to stay within the output $35\,\mathrm{ns_{rms}}$ specification. Therefore, the residual jitter will be caused by the uncompensated portion of the deterministic $\Sigma\Delta$ jitter and by the native stochastic jitter from the oscillator.

Introducing a variable delay requires a *Digital-to-Time Converter* (DTC) [40]. In their simplest version these circuits can implement the delay as a voltage ramp, created for instance as a capacitor charged with a constant current, that crosses a threshold after a definite time. Building an array of selectable capacitors and/or current sources forms a basic DTC.

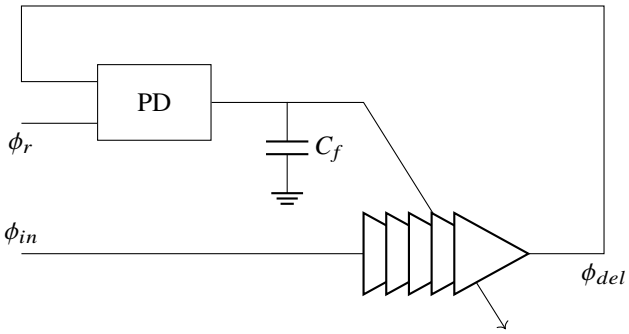


Figure 4.21: *Generic scheme of a DLL. A Phase Detector (PD) outputs a current that, integrated on a capacitor filter, tunes the delay of a delay line to match the input reference period.*

Nevertheless, a very precise gain, or equivalently Full-Scale Range (FSR)[1], is necessary in order to reach a satisfactory compensation. In fact, the jitter compensation scheme, with different gain errors, was simulated in MATLAB: the result is shown in Figure 4.20. A FSR error of just $1\,\text{LSB} = \frac{T_0}{2^B} = 58\,\text{ns}$ is enough to increase the output jitter to 42 ns, over the specification.

Unfortunately, the only way to set precisely the gain of the "voltage-ramp" DTC is an initial calibration. This is in contrast with the proposal of the previous chapters of building a RTC as uncalibrated as possible, to exploit the high repeatability of polysilicon MEMS devices and propose a concept for a low-cost RTC.

Therefore, a better DTC implementation consists in adopting a chain of delay elements whose combined delay is controlled in feedback by a servo-loop, which forms a Delay-Locked Loop (DLL) [40]–[42].

4.4.1 DLL Description

A generic block scheme of a DLL is shown in Figure 4.21. A reference signal ϕ_r constitutes the input of a Phase Detector (PD), whereas signal ϕ_{in} is the input of a delay chain, formed by N_{dc} delay cells. Often, ϕ_{in} and ϕ_r are the same signal. The output of the delay chain ϕ_{del} is a delayed copy of ϕ_{in} and is compared against ϕ_r by the PD, which outputs an error signal proportional to time difference between the edges of its two inputs. The error signal is then filtered, by capacitor C_f in this example, and used to

[1] A DTC is nothing but a Digital-to-Analog Converter (DAC) where the analog variable is a time, rather than a voltage or a current. Therefore, the usual terminology of DACs (Differential/Integral Non-Linearity, Full-Scale Range. . .) will be used.

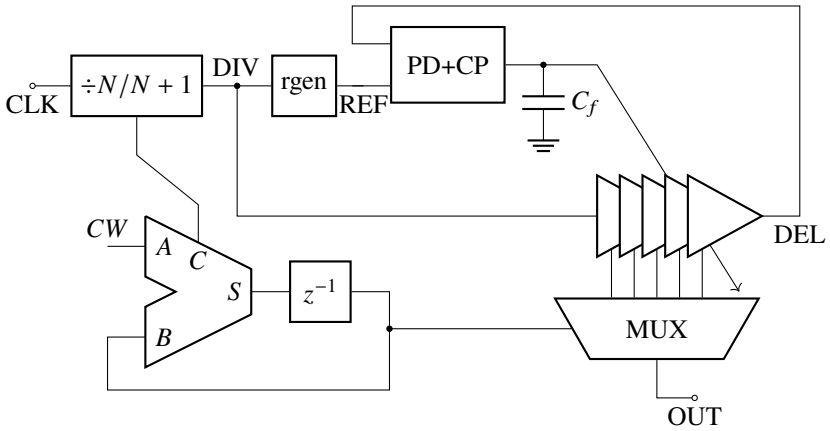


Figure 4.22: *Block scheme of the compensation circuit highlighting the internal structure of the DLL. The MC and MUX select signals are synchronized by additional flip-flops. To compensate the jitter on both the rising and falling edges, there are two DLL loops in parallel that share the same delay line. In particular, there are two PD+CP, C_f and "rgen" blocks, whereas the rest of the circuit is not replicated.*

control the total delay applied by the delay line. In this way, at steady-state ϕ_{del} is a squarewave synchronized with ϕ_r.

The outputs of each delay cell, that will be called $\phi[1], \phi[2], \ldots, \phi[N_{dc}]$, are useful side-products of this operation. Defining T_r as the period of ϕ_r, they are N_{dc} replicas of ϕ_r that are all spaced by an equal time $\Delta t = \frac{T_r}{N_{dc}}$ with respect to the previous one.

These N_{dc} clock phases are what is needed for the compensation scheme proposed, that is represented in Figure 4.22. The delay chain can introduce a variable delay according to which clock phase is selected by means of a multiplexer driven by the SUM signal of the accumulator and the DLL acts as a servo-mechanism to adjust real-time the FSR of the DAC. The overall compensation system is illustrated in Figure 4.22. The divided signal DIV drives the delay chain and is used to generate the reference signal REF that goes to the PD.

The choice of the reference signal REF has a big impact on the overall design. Since the full-scale delay of the DTC should be $FSR = T_0$, the most obvious choice for REF is using the signal of the reference oscillator. In this way, the FSR would be matched to the reference period T_0 and a chain of just $N_{dc} = 32$ delay cells would provide the required DTC resolution. Nonetheless, clocking a 32-elements delay chain at 540 kHz would consume too much, with a simulated consumption close to the total current budget of 1 µA.

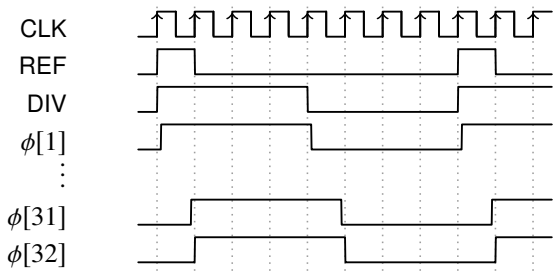


Figure 4.23: *Timing diagram of the digital signals employed in the DTC. For illustration purposes a periodicity of 8 rather than the actual 16 is used.*

An alternative choice for REF could be, instead, the divided signal DIV[2], so that now the delay chain has a slower clock frequency. But this choice has little to no advantage in terms of consumption. In fact, the reference period has increased by a factor of 16 and, to reach the same time resolution, the required N_{dc} has also increased by a factor of 16, so that now a $N_{dc} = 32 \times 16 = 512$ would be necessary. In the end, the dynamic consumption of the delay line sees a reduced switching frequency but a correspondingly increased total capacitance to switch, so that no power consumption reduction is achieved.

A smarter choice consists, instead, in producing a reference signal REF like the one shown in Figure 4.23. REF is a signal synchronous with DIV, i.e. their rising edges coincide, but REF has a high duration of only T_0. The PD will then compare the time difference between the rising edge of $DEL = \phi[N_{dc}]$ with the falling edge of REF. This choice:

- achieves the required condition $FSR = T_0$.
- does so by using the minimum N_{dc}, with beneficial effects on the power consumption and the area on silicon.
- does so using the slower clock frequency, i.e. 32.768 kHz, thereby minimizing the power consumption.

Therefore, this choice achieves the pros of the two previously mentioned solutions without their disadvantages. Since the multi-modulus divider is nothing but a down-counter that is reset to 15 or 16 once it reaches the state where all FF have a low output, the 5 FF outputs can be given as inputs to a 5-input NOR gate (the block indicated as "rgen" in Figure 4.22): the result is the desired REF.

[2]Actually, since the DIV period is variable between $T_0/16$ and $T_0/17$, an auxiliary waveform with period $T_0/16$ should be derived.

Transistor	Size [μm / μm]
M1	$(20/1.7) \times 4$
M2	$(2/0.13)$
M3	$(1/0.13)$
M4	$(6/4) \times 2$
M5	$(2/0.5)$
M6	$(1/0.5)$

Table 4.4: *Transistor-level sizing of the delay cell element.*

To conclude, it was chosen to compensate both the rising and the falling edges of the DIV clock. To do so, the scheme of Figure 4.22 was partially replicated, even though it is not shown in the figure for the sake of simplicity. Specifically:

- The delay line has two control inputs that control its delay *independently* on the rising and falling input edges.
- The two control inputs are set by two independent phase detectors, that receive two different REF signals, that are replicas shifted in time produced by separate "rgen" blocks.
- The delay line is not duplicated, as well as the multiplexer. Since the rising and falling edges of DIV are affected by the same jitter, the same SUM signal can select the same $\phi[k]$ for two adjacent time stamps.

4.4.2 DTC Design

Once the general architecture of the DTC has been introduced, the physical design of the converter will be described. All the blocks have been designed with GO1 transistors to reduce the dynamic consumption. The only exception is the Charge Pump, an analog block whose consumption is mainly static, that was designed in GO2 to exploit their lower flicker noise.

Delay Chain

The delay cell used within the DTC is based on a current-starved inverter buffered by another inverter, a solution used for instance in [43]. The circuit is shown in Figure 4.24 and features two current-starving transistors for controlling the delay on both the rising and falling edges, driven by the input tuning voltages $V_{t,r}$ and $V_{t,f}$. A tri-state inverter loads every delay cell: this builds the base of a binary selection tree of 63 tri-state inverters that picks the correct clock phase, according to the SUM signal.

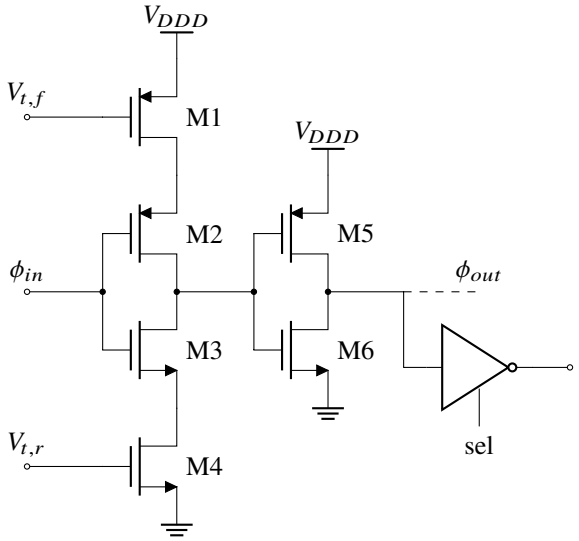


Figure 4.24: *Schematic of the delay cell, built from a current-starved inverter buffered by another inverter. Voltages $V_{t,r}$ and $V_{t,f}$ control, respectively, the rising and falling edge delay. The following delay cell and the tri-state inverter used to select the specific phase clock contribute to the load capacitor of the cell.*

The sizing of the transistors was driven mainly by the trade-off between the matching of the delay introduced by the 32 delay cells, which improves for large components, and the minimization of the power consumption, which pulls towards smaller dimensions. All the transistors were GO1 and the final sizing is reported in Table 4.4. The delay matching, simulated with Monte-Carlo simulations, has a standard deviation of 4.8 % for the delay on the rising edge and 2.8 % for the falling edge.

The chain was also sized in order to have a nominal control voltage roughly at half the supply voltage of 0.8 V. In the end, the resulting voltage-to-delay gain of the stage is $23\,\mu s\,V^{-1}$.

The average consumption, obtained with a post-layout transient simulation, is 86 nA, subdivided in 33 nA for the delay line and 53 nA for the tri-state inverter selection tree.

The accumulated noise introduced by the total delay chain is negligible amounting to an RMS value of 4 ns.

Phase Detector and Charge Pump

In conventional DLL circuits, a Phase Detector (PD) measures the phase (or time) error between a reference signal and the delayed signal, so that a

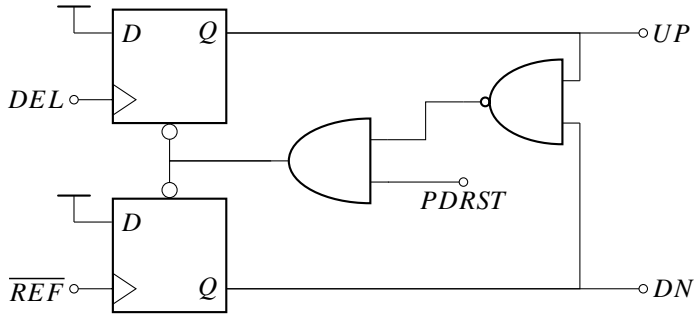


Figure 4.25: *Schematic of the used phase detector, which receives as input $\overline{REF}$ to match the delay edge on the falling edge of REF. The PDRST input is needed to prevent the "stuck at minimum delay" issue.*

Charge Pump (CP) can inject or remove a charge proportional to this error from the loop filter, often formed by just a capacitor C_f. Often, the PD is formed by a Set-Reset latch (SR) [44] or equivalent topologies. The arrival of the reference edge sets the latch – or resets, according to the correct polarity to achieve negative feedback – whereas the arrival of the delayed edge resets it. Assuming a 50 % duty cycle, this means that at steady state the reference and delayed signals will be locked with a 180° phase shift. In this situation, in fact, there is no net charge drawn each cycle from C_f, as a certain amount of charge will be injected for half of the reference period and the same amount will be removed in the other half of the period.

Nevertheless, in the present case this PD implementation is not suitable, since the duty cycle of REF is very different from 50 %. Therefore, a PD similar to the Phase-Frequency Detector (PFD) very popular in Analog Phase-Locked Loops (PLL) [45] was used. In this circuit, two Data-Trigger FFs are set by, respectively, a REF edge and a DEL edge. The outputs of the two FFs encode the UP/DOWN signals that, when asserted, drive the CP to increase/decrease the charge on C_f with a fixed current for an amount of time proportional to the high duration of these control signals. As soon as the state $UP = 1$, $DOWN = 1$ is entered, a combinational reset network brings the state back to the state $UP = 0$, $DOWN = 0$, so that no current is either injected or removed from C_f. Therefore, only 3 states are available, as schematically shown in Figure 4.26:

1. $UP = 0$, $DOWN = 0$: no charge is injected or removed from the C_f.
2. $UP = 1$, $DOWN = 0$: charge is injected into C_f.
3. $UP = 0$, $DOWN = 1$: charge is removed from C_f.

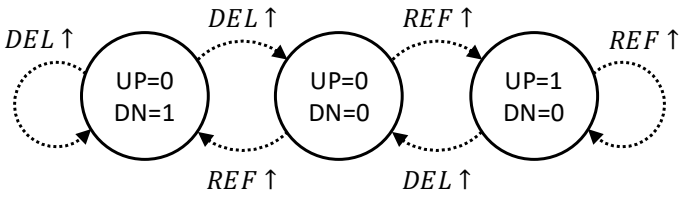


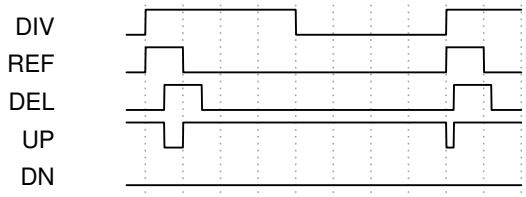
Figure 4.26: *State diagram of the Phase-Frequency Detector used.*

At steady state the voltage on C_f is constant: this can be true only if the edges of REF and DEL are perfectly aligned, with no requirement on their duty cycle.

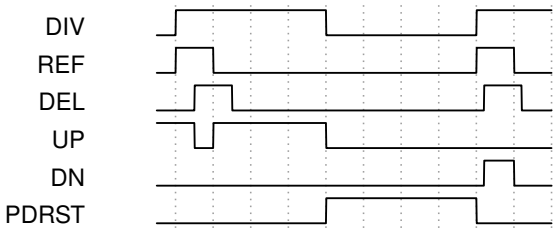
Nonetheless, if any unpredictable event, e.g. at the system startup, initializes the PFD state machine in the wrong state the system may remain in a non-desired situation [46]. Let us consider the state diagram of the PFD phase detector shown in Figure 4.26 and suppose that the system is initialized with the delay chain in a minimum delay condition and the PFD is initialized in the $UP = 1$, $DOWN = 0$ state, where UP causes the delay to decrease. In this case, represented in Figure 4.27a, the DEL edge occurs before the REF edge and therefore the PD should provide a DN pulse. What happens, instead, is that the system goes briefly in the $UP = 0, DN = 0$ state before returning to the initial $UP = 1$, $DN = 0$ state. The delay is not increased and a proper DLL operation is never achieved; other similar examples could have been made.[3] To overcome this problem, the PDRST signal shown in Figure 4.25 was introduced: this signal resets the PFD state machine to the $UP = 0$, $DOWN = 0$ state while DIV is low (see Figure 4.27b). This prevents any initialization problems at startup and prevents unpredictable events to keep the DLL in a false lock condition indefinitely.

The charge pump circuit is shown in Figure 4.28. The design, similar to the one proposed in [47], derives the voltage references from the input current I_{cp} and pumps an output current generated by M7 or M6, according to which MOS is enabled by the source switches M8/M5. M5/M8 are switched by digital inputs UP/DN: these elements are cascoded by transistors M6/M7, so that spurious charge injection are partially decoupled from the output capacitor. Nevertheless, in the simulation phase a fine tuning was necessary to avoid injecting a net charge Q_{os} at each cycle that corresponds

[3] Contrarily, a PLL can cope with this situation, thanks to the possibility to have the so-called *cycle slips* in the VCO.



(a) *In this case, when no counter-measurement is used, the false initialization leaves the system stuck in a situation where $UP = 1$ for most of the reference period and $DN = 0$ always. The correct situation would be to have short pulses of DN to slowly increase the delay but the system moves in the opposite direction.*



(b) *Given the same initial conditions, in this case when DIV goes low, PDRST resets the PFD, so that at the second phase comparison a short DN pulse is produced, restoring the correct behavior of the DLL.*

Figure 4.27: *Illustration of the "stuck at minimum delay" problem that can happen when using a PFD if a wrong initialization or an unpredictable event occur.*

Transistor	Size
M1	$(2/0.5) \times 2$
M2	$(6/1) \times 2$
M3	$(2.75/1) \times 2$
M4	$(1.7/0.5) \times 2$
M5	$(2/0.5) \times 2$
M6	$(6/1) \times 2$
M7	$(2.75/1) \times 2$
M8	$(1.7/0.5) \times 2$
M9	$(2.75/1)$
M10	$(1.7/0.5)$

Table 4.5: *Sizing of the CP circuit used in the DLL loop.*

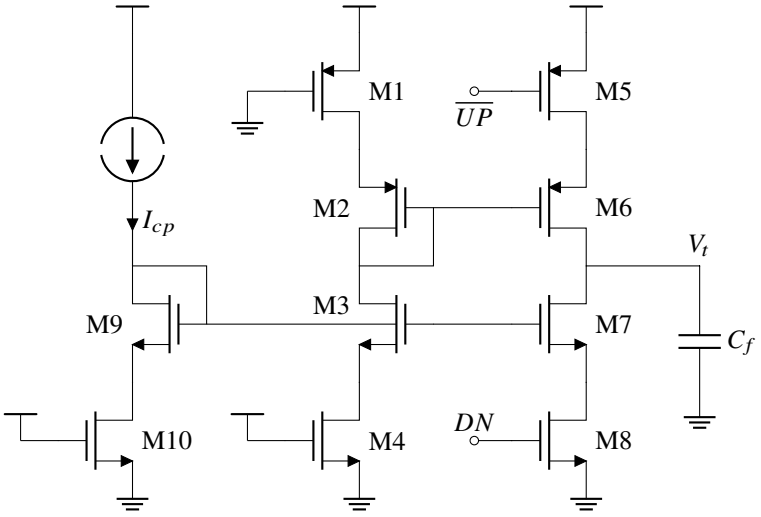


Figure 4.28: *Charge Pump schematic. The output branch composed by M5, M6, M7, M8 is replicated to control the delay on both edges, whereas the other two branches are shared between the two loops. The network that generates the reference current I_{cp} adds an additional branch, so that the total static consumption of the CP is $I_{cp} \times (1+1+1\times 2) =$ 80 nA.*

to an offset of the charge vs time difference characteristic of the CP. The importance of this derives by the fact that a fixed charge injected at each phase comparison operation is directly referred at the input of the DLL as a steady state time error, i.e. an error on the FSR of the DTC indicated as FSR_ϵ, divided by the CP gain $\frac{I_{cp}}{2\pi}$:

$$FSR_\epsilon = Q_{os} \cdot \frac{2\pi}{I_{cp}} \tag{4.22}$$

A low FSR_ϵ is clearly in trade-off with a low consumption due to the bias current I_{cp}, which was set to 20 nA. The transistor sizing of the circuit is reported in Table 4.5: the overall DC current consumption of the CP is of 60 nA overall, considering also the bias branch.

Noise and Consumption

Overall, the current consumption of the compensating scheme is subdivided as indicated in Table 4.6.

At this point, it is interesting to make a comparison between the proposed jitter suppression strategy and the conventional one, that employs a PLL as a phase filter. Specifically, this comparison will be done by comparing the noise/jitter versus power trade-off.

Block	Consumption [nA]
DDSM + Divider	32
PD (×2)	34
CP	60
Delay Line	33
Selection Tree (Tri-state inverters)	53
Total	212

Table 4.6: *Consumption of the compensation scheme composed by the fractional divider and the DTC split in the fundamental blocks.*

A DLL and a PLL share a quite similar topology. Both have an input phase detector that produces the error signal and a loop filter that produces the tuning signal that controls the voltage-controlled oscillator (in a PLL) or a variable-delay element (in a DLL).

Therefore, a strategy that optimizes the noise and power consumption of the phase detector or the loop filter will be beneficial in roughly the same way in both solutions. Then, it is interesting to compare the two strategies in the two blocks that set apart the two architectures: the VCO and the delay chain.

The bandwidth of a PLL is constrained to be much smaller than the reference frequency, to filter as much as possible the very large noise, shaped at high-frequency due to the $\Sigma\Delta$ modulation, introduced by the fractional divider. As a result, the high-pass filtered VCO noise will appear at the output of the system practically unfiltered.

To compute the period jitter of a generic ring VCO – the only possible choice at such low frequencies – it is convenient to resort to the approximation derived in [48, p. 1808, (32)]:

$$\sigma_j^2 = \frac{kT}{I f_0}\left[\frac{2}{V_{DD}-V_{TH}}\left(\gamma_n+\gamma_p\right)+\frac{2}{V_{DD}}\right] \tag{4.23}$$

By fixing a certain σ_j jitter requirement, one finds that the required current I_v that the VCO requires to reach the σ_j jitter is:

$$I_v = \frac{kT}{\sigma_j^2 f_{rtc}}\left[\frac{2}{V_{DD}-V_{TH}}\left(\gamma_n+\gamma_p\right)+\frac{2}{V_{DD}}\right] \tag{4.24}$$

It is to be noted that I_v does not depend on the VCO running frequency: if a M times larger VCO frequency is used, the frequency division in the feedback path of the PLL accumulates the jitter for M periods, thus canceling the factor M from the equation.

Block	Area [μm^2]
Oscillator	200×80
Divider and Modulator	80×90
DTC	100×175
Bias Network	60×45
Total	290×240

Table 4.7: *Area occupation by functional block.*

The noise introduced by the delay chain can be described by the same formula, since in the derivation of (4.23) the fact that the inverter chain is closed in loop does not affect the period jitter estimation. Since the delay chain applies an overall delay $\tau_{FSR} = \frac{1}{f_0}$, if it were closed in a loop to create a ring VCO, it would oscillate at a frequency of $\frac{1}{2\tau_{FSR}} = \frac{f_0}{2}$. Therefore, the application of (4.23) to the DLL case needs using $\frac{f_0}{2}$ as frequency, resulting in a current I_d required by the delay chain of:

$$I_d = \frac{kT}{\sigma_j^2 \frac{f_0}{2}} \left[\frac{2}{V_{DD} - V_{TH}} \left(\gamma_n + \gamma_p \right) + \frac{2}{V_{DD}} \right] \cdot \frac{f_{rtc}}{f_0} \tag{4.25}$$

which has been corrected by a factor $\frac{f_{rtc}}{f_0}$ considering that the delay chain actually has to consume this amount of current for just a fraction of the period, as it is evident by looking at the waveforms of Figure 4.23.

In the end, by adopting a DLL there is an advantage over a PLL solution by a factor of:

$$\frac{I_v}{I_d} = \frac{1}{2} \left(\frac{f_0}{f_{rtc}} \right)^2 = 135 = 21 \text{ dB} \tag{4.26}$$

Actually, this improvement is optimistic, since the comparison is made only in terms of noise vs power. The fact that this delay chain is used as a DTC imposes a minimum matching, that can erode this advantage. A better matching will require larger components so that a larger average current will be drawn from the supply for the same full-scale delay. In fact, Abidi's model would predict a consumption of 12 nA for the delay chain, given the simulated jitter of ≈4 ns: the present implementation shows a delay line consumption increased by a factor of 3 (or 4.7 dB) with respect to a theoretical minimum, still keeping a significant advantage when compared to a PLL implementation.

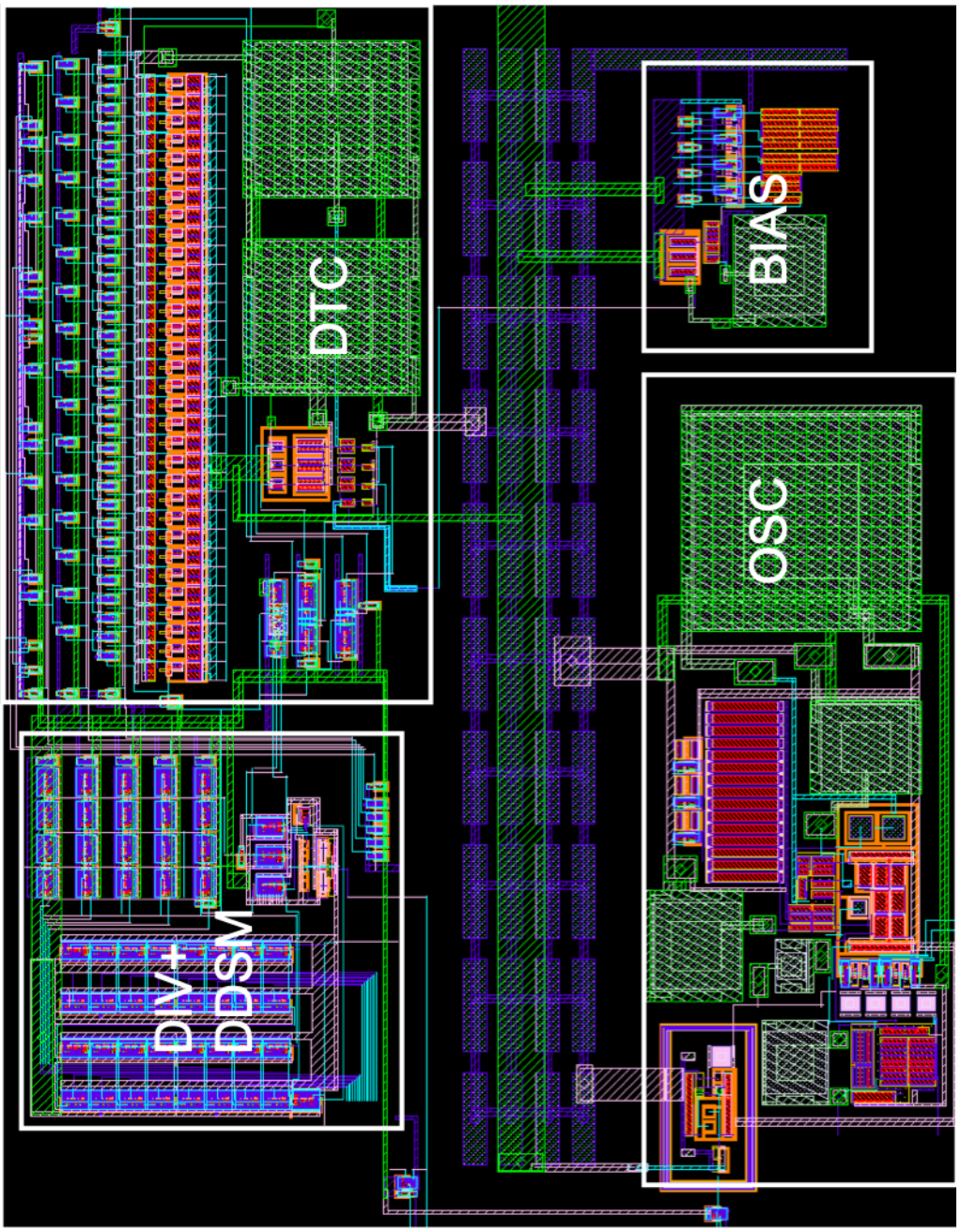


Figure 4.29: *Layout of the chip, which totals an area occupation of* $290\,\mu\text{m} \times 240\,\mu\text{m}$.

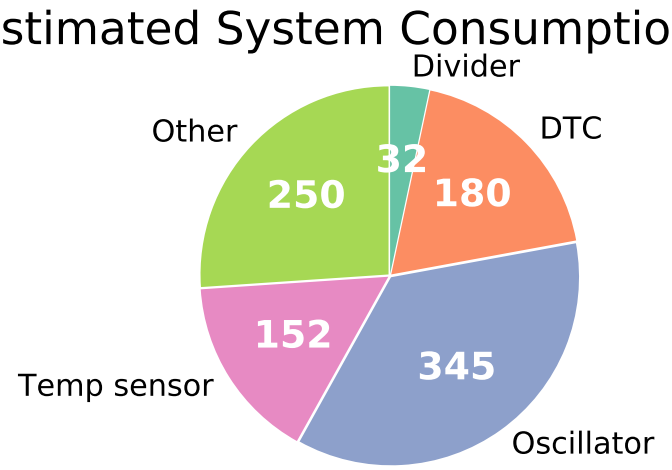


Figure 4.30: *Consumption per block of the whole integrated system. An estimate for the consumption of the temperature sensor was made by using the* 38 nJ/conversion *of the temperature sensor used in [17] divided by the* T_{conv} = 250 ms *obtained in Chapter 3. An approximate consumption of* 300 nA *was added to the pie chart to account for additional auxiliary blocks not implemented, such as the charge pump to produce the rotor voltage, bias networks, supply filtering. . . to reach a total consumption of* 959 nA.

The layout of the blocks described in the present Chapter is shown in Figure 4.29. The total used area amounts to 290 µm × 240 µm, the area occupied by the single blocks is mentioned in Table 4.7.

Figure 4.30 shows the subdivision of the current consumption between the blocks of the system. For completeness, an estimation of the consumption of the temperature sensors and other block not realized is made, for a total consumption of 959 nA.

CHAPTER 5

Experimental Results

The Chapter describes the setup to test the RTC. A custom board was designed to communicate with the Integrated Circuit through its on-chip SPI interface and with a PC through a USB connector. The board is conceived to operate inside a climatic chamber at temperatures as high as 85 °C *and measures the chip temperature with an on-board temperature sensor. The measurement results confirm the repeatability of the Temperature Coefficient of Frequency of polysilicon devices, confirms the expected current consumption of the implemented blocks and shows a long-term frequency stability in a long-running measurement.*

5.1 Characterization Setup

The IC described in Chapter 4 is also equipped with a Serial-Peripheral Interface (SPI) interface, to provide the possibility of trimming several parameters, specifically:

- Capacitance C_d, responsible for the attenuation of the drive voltage of the MEMS resonator.
- Resistance R_3 , able to modify the startup current of the oscillator

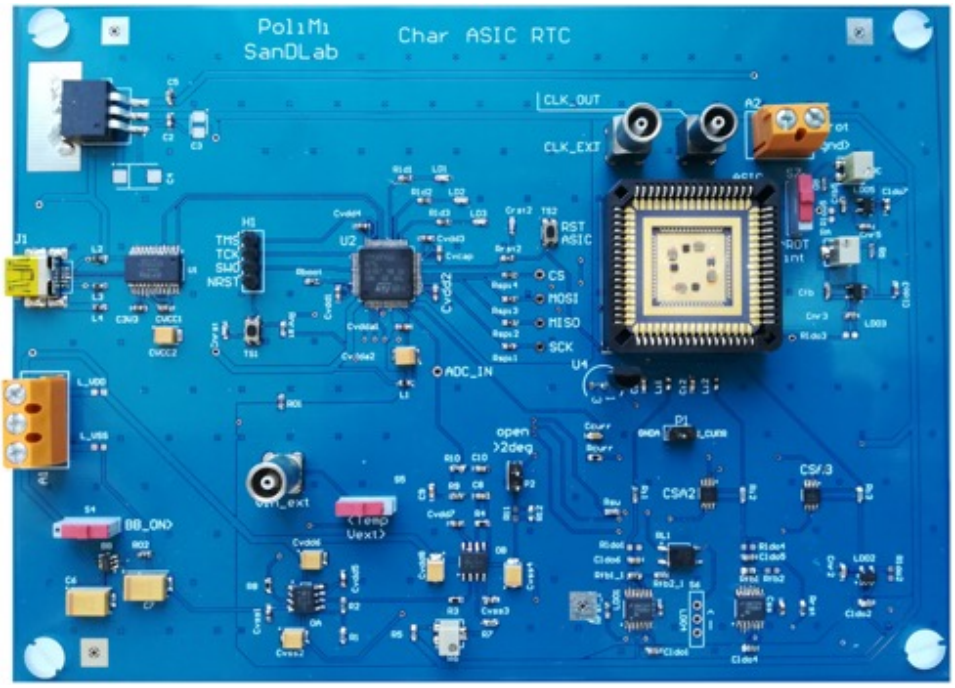


Figure 5.1: *Picture of the designed board. Four pairs IC+MEMS are bonded on a CLCC68 ceramic carrier; each can be tested by rotating the carrier inside the on-board socket.*

and, partially, the steady-state voltage amplitudes within the oscillator itself.

- The Command Word CW, that changes the average division factor of the divider.
- The clock signal to output: the squarewave 540 kHz oscillator clock, the divided clock DIV and the compensated output of the DTC are available.
- The charge pump current I_{cp}, that can reduce the FSR error of the DTC if increased.

Therefore, a characterization board able to manage the required digital communications was developed. A picture of the PCB is shown in Figure 5.1.

The supply voltage of the chip is split into multiple PADs:

1. A 1.2 V supply for the oscillator block.
2. A 0.8 V supply for the fractional divider and the compensating DTC.
3. A 1.2 V supply for the SPI block.

The rotor voltage of 2.4 V is also generated on-board. These supplies are generated on board by 4 Low-Dropout Regulators (LDO), whereas the whole board takes as supply the USB 5 V supply.

The average current flowing into the supply PADs is read with a 10 kΩ resistor in series, whose voltage is read with a MAX9934T Current Sense Amplifier (CSA), whose gain is 25 $\mu A\,mV^{-1}$. The output current produced by these amplifiers is then read on a 4 kΩ load resistor.

An STMF411RE microcontroller was used to write the parameters over the SPI interface and communicate with the user through the on-board FT232 UART↔USB converter. Its internal Analog-to-Digital Converter (ADC) is used to digitize the current consumption as is read by the CSAs and to measure the temperature measured by an LM35 sensor, placed as close as possible to the MEMS resonator.

A Keysight 53230A frequency counter was used to measure the output frequencies and jitter. Each jitter measurement was performed on 10 000 samples.

5.2 TCf Characterization

The board was placed inside a climatic chamber (Figure 5.2) and the measurement were performed with several sweeps of the temperature between

Figure 5.2: *The board – in these case two identical boards to perform the measurements on two devices within the same temperature sweep – is placed inside a climatic chamber. The temperature was swept between* 5 °C *and* 85 °C *to avoid problems due to the condensation of water vapor inside the chamber. On the right it is possible to see the bench-top instruments used for the measurements.*

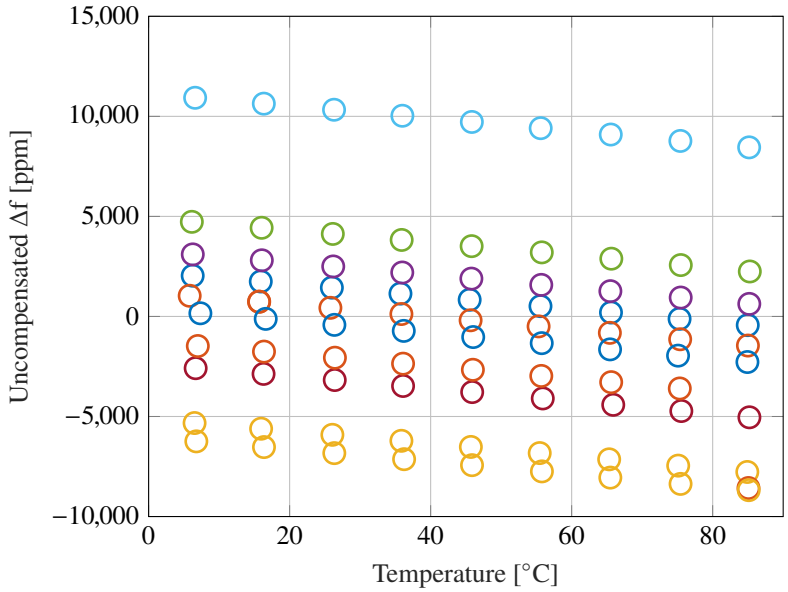


Figure 5.3: *Oscillator frequency variation versus temperature. The traces are all normalized to the same nominal frequency of* 600 kHz*, so that the relative spread of* f_0 *over the population of the measured devices is visible.*

5 °C and 85 °C. Temperatures below 0 °C were not explored to avoid any problem due to the condensation of the water vapor inside the chamber. An automatic MATLAB routine interacting with the instruments and the on-board microcontroller measured the main parameters of the system, namely:

- The current temperature
- The oscillation frequency f_0
- The consumption of the analog portion of the system, from the 1.2 V supply.
- The consumption of the digital portion of the system, from the 0.8 V supply.
- The jitter of the oscillator.
- The output jitter after the compensation at the DTC output.

Since the devices were fabricated with a new etching machine, the lack of past experience caused the resonators to be fabricated with a resonance frequency around 600 kHz. This fact does not invalidate the results that

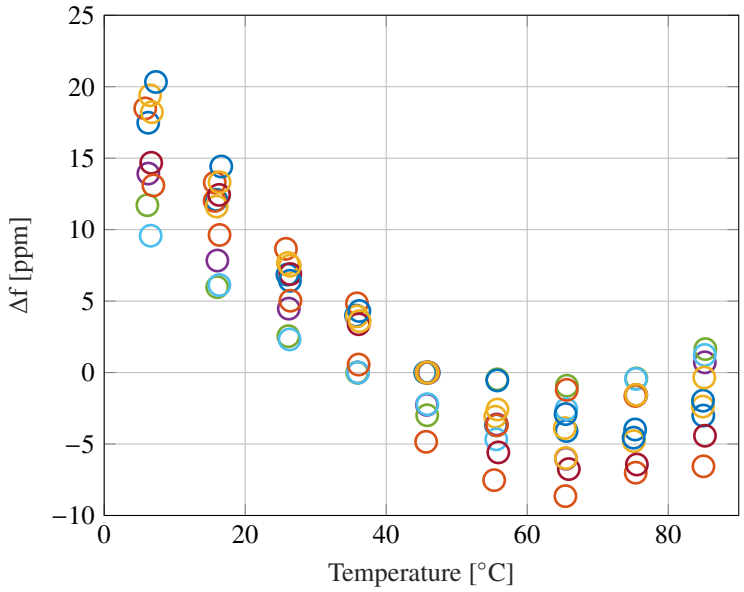


Figure 5.4: *Mathematical compensation of the frequency drift with the average TC f_1 and TC f_2 obtained in [2]. The trend is not completely compensated, which may be due to an improved thermal stability of the chamber used in this second experiment, to non-linear phenomena or both. Nonetheless, it is to be noted that the residual drift stays within the measurement noise/uncertainty level of* 30 ppm *(see Figure 2.9 at page 30 for a comparison).*

will be exposed, since they could be repeated also at the target frequency of 540 kHz. Figure 5.3 shows the measured $f_0(T)$ dependence, normalized to a reference of 600 kHz, measured on the available samples. A theoretical compensation, like the one performed in Chapter 2, is shown in Figure 5.4. With the same $TCf_1 = -28.9\,\mathrm{ppm/K}$ and $TCf_2 = -24 \times 10^{-3}\,\mathrm{ppm/K^2}$ used in [2], the compensation shows an evident uncompensated residual drift.

A first comment about this result is that the repeatability of the thermal properties is striking: the tighter-gap devices coupled to the developed IC were fabricated about 18 months after the ones for which the thermal coefficients were determined and with different machinery. Still, an overall spread of only 30 ppm is obtained with the same compensating coefficients.

Secondly, assuming that the residual quadratic drift is not to be ascribed to the mechanical devices, other causes may be responsible for it. In particular, this second run of measurements used a different climatic chamber, for practical reasons. This chamber showed a much better thermal stability, that allowed a better thermal settling. Therefore, the temperature measured from the on-board temperature sensor is a more accurate proxy of the temperature of the nearby MEMS device. In fact, this residual information on the $f_0(T)$ could have been – at least partially – hidden under the ±20 ppm measurement uncertainty. Another possible cause may be a slightly larger excitation of the non-linearity of the resonator, that introduced a modification of the $f_0(T)$ in linear regime.

All in all, Figure 5.5 shows the same operation repeated with the average TCf_1 and TCf_2 recomputed for this second batch of resonators:

$$TCf_1 = -29.56\,\mathrm{ppm/K} \tag{5.1}$$

and

$$TCf_2 = -19.22 \times 10^{-3}\,\mathrm{ppm/K^2} \tag{5.2}$$

The repeatability in the $\frac{\Delta f}{f}(T)$ of this oscillators is within ±5 ppm across the 5 °C to 85 °C range, potentially still limited by the precision of the setup.

5.3 Consumption and Jitter

The current consumption and jitter of the oscillator was measured as a function of temperature and is reported in Figure 5.6. The average consumption measured at room temperature across 7 samples is 410 nA and a standard deviation of 86 nA. The average consumption is 19 % higher than what was expected in simulation, and also the dispersion is higher, even though measured on a limited number of samples.

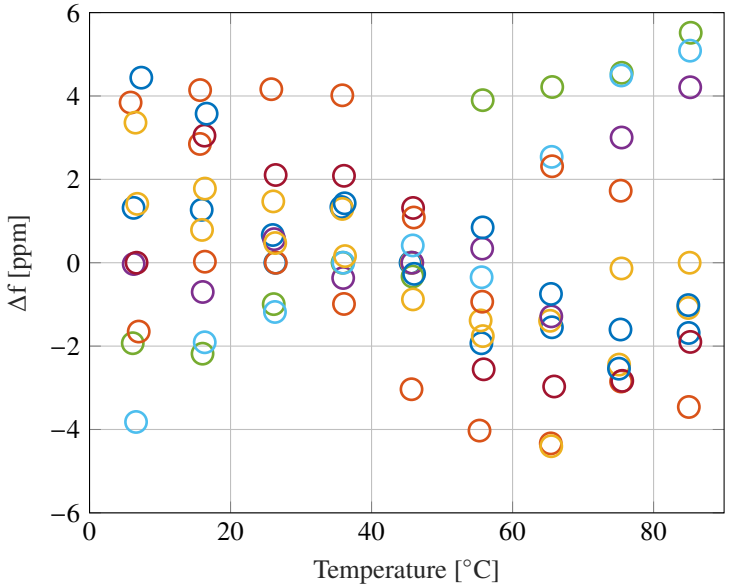


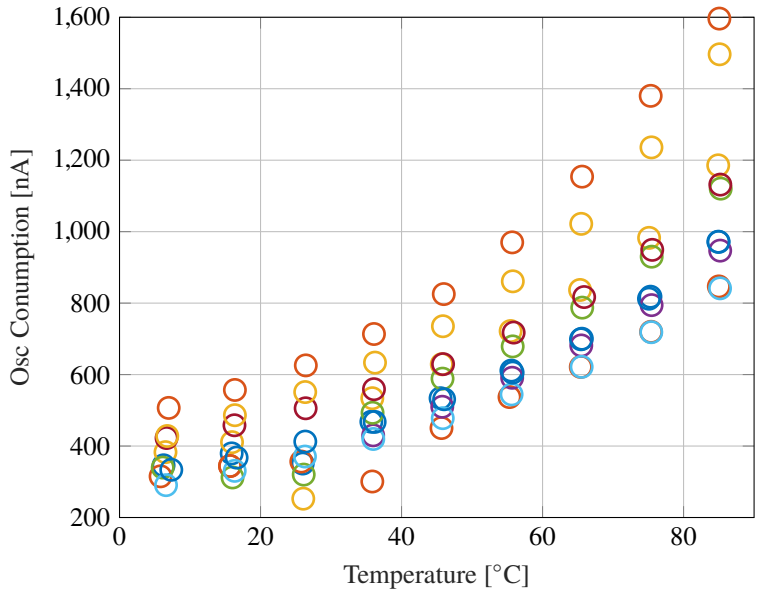
Figure 5.5: *Mathematical compensation of the native frequency drift.*

The average value is for sure increased with respect to the simulation results by the 10 % higher resonance frequency of the MEMS. Since the minimum achievable current consumption is proportional to the oscillation frequency squared, this corresponds to an equivalent 20 % increased consumption. The increased standard deviation can be partially due to a spread during the etching phase of the MEMS resonators, which impacts the gap dimension and therefore the transduction efficiency and the equivalent motional resistance.

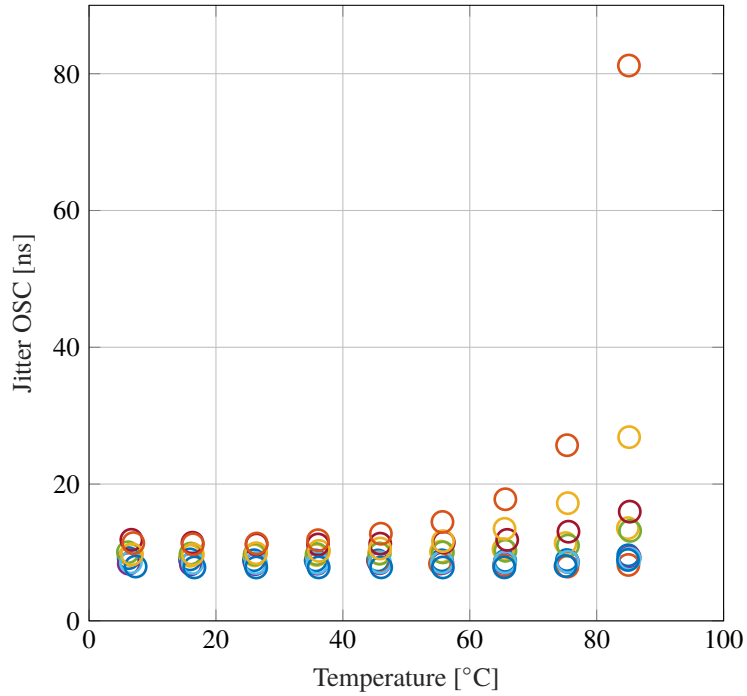
Unfortunately, it was not possible to have direct access to the equivalent electrical parameters. Nevertheless, the resonant frequency f_0 is correlated with the etching spread as well, since the etching has a direct impact on the equivalent stiffness k and on the modal mass m. Given an unknown critical dimension loss x, one can write that the consumption of the oscillator I is proportional to:

$$I \propto R_m \omega_0^2 \propto \frac{1}{\eta^2} \cdot k \propto g^4 \cdot W^3 = (g + x)^4 \cdot (W - x)^3 \tag{5.3}$$

where η is the transduction factor, k the mechanical stiffness, g the gap and W the width of the resonant beam. By differentiating, it is possible to arrive



(a) *Consumption measurements.*



(b) *Jitter measurements.*

Figure 5.6: *Jitter and current consumption measurement on the RTC samples. The curves that show the highest jitter and the highest consumption refer to the same sample.*

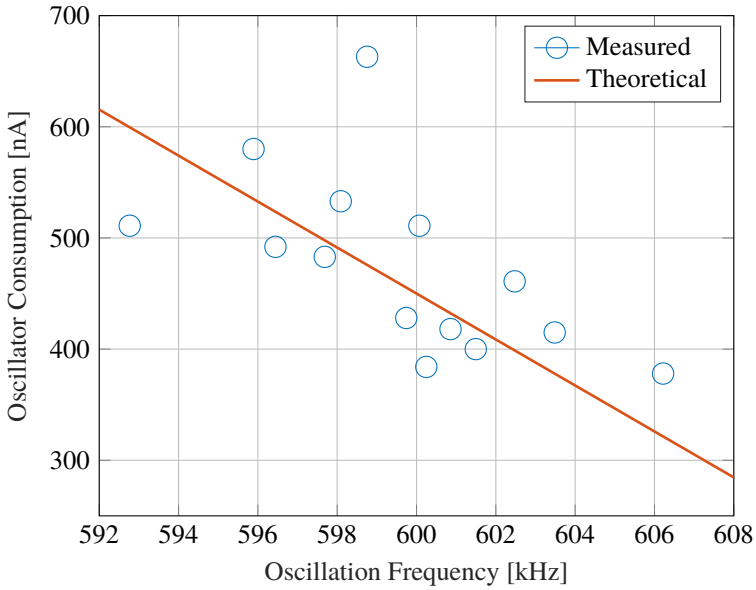


Figure 5.7: *Scatter plot of the measured oscillator consumption versus the oscillation frequencies. From the correlation, even though a partial one, one can see that a good fraction of the consumption spread is due to a spread on the etching process during the fabrication of the MEMS.*

to the following expected correlation between x and I_{osc}:

$$\frac{\Delta I}{I} = 4\frac{x}{g} - 3\frac{x}{W} = 4\frac{W}{g} \cdot \frac{x}{W} - 3\frac{x}{W} \tag{5.4}$$

Since this correlation is not observable through measurement, one can combine (5.4) with the relationship that links f_0 and x:

$$f_0 \propto k \propto (W - x)^{\frac{3}{2}} \Rightarrow \frac{\Delta f}{f} = -\frac{3}{2} \cdot \frac{x}{W} \cdot \gamma \tag{5.5}$$

where an additional factor $\gamma \approx 2$ takes into account the effects of the slots, according to internal discussions with the mechanical designers involved in the project. Then, one obtains:

$$\frac{\Delta I}{I} = \left(\frac{4}{3} \cdot \frac{W}{g} - 1\right) \frac{\Delta f}{f} \tag{5.6}$$

which is observable through measurement and is plotted in Figure 5.7 (solid line) with $W = 15\,\mu m$ and $g = 700\,nm$. The result matches quite well the measurement result, which confirms that a fraction of the dispersion of the oscillator consumption is due to the MEMS. The residual spread around the theoretical curve of Figure 5.7 confirms the deviation due to the IC only obtained through simulation.

The measured consumption of the digital blocks from the 0.8 V supply is shown, instead, in Figure 5.8. Since it is not possible to discern the different contributions to the total consumption from the digital supply, the static contribution and the dynamic contribution could be separated by providing an external reference clock from the outside, bypassing the oscillator. The result is shown in Figure 5.9, where the external clock frequency is varied between 100 kHz and 800 kHz. A 0 Hz clock – or a very slow one – would be useful to know precisely the DC consumption of the bias references of the Charge Pump inside the DLL, but this measure is not reported because the dynamic flip-flops would be refreshed rarely enough to enter undefined states with additional and meaningless consumption. Nonetheless, by interpolating the available points, it is obtained that the average current drawn from V_{DDD} due to static consumption is 90 nA, of which 55 nA are to be ascribed to the CP itself, whereas the rest is due to the bias network that, in a final implementation would be shared by several blocks, such as the analog temperature acquisition chain. Moreover, from simulation 20 nA are consumed by the portion of the oscillator output comparator that acts as a 1.2 V to 0.8 V level-shifter.

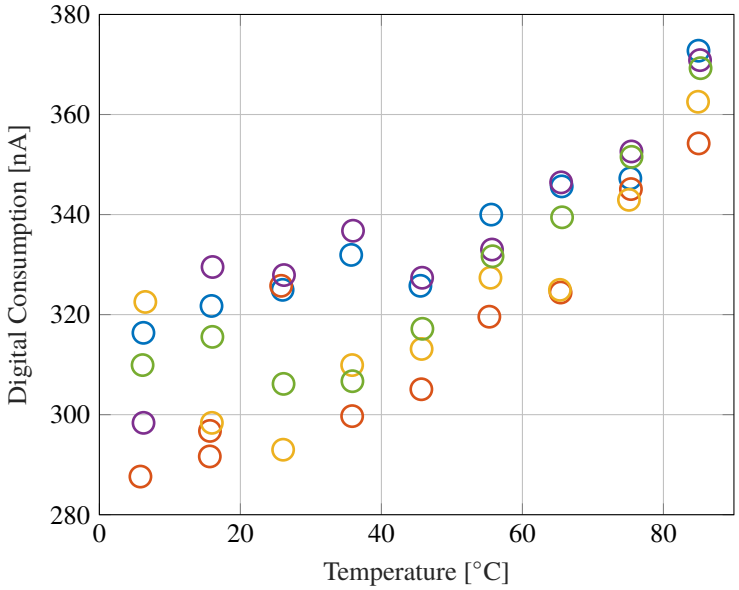


Figure 5.8: *Consumption measured for the digital blocks, i.e. the fractional divider, the DDSM and the DTC.*

The dynamic portion of the consumption, measured with the target 540 kHz input clock frequency, out of the 252 nA total would therefore be:

$$I_{dig,dyn} = I_{dig,tot} - I_{dig,DC} - I_{comp} = 252\,\text{nA} - 90\,\text{nA} - 20\,\text{nA} = 142\,\text{nA}$$

thus substantially validating the expectations from simulation. The residual output jitter was measured applying a Command Word CW = 0xAAAA, a value chosen randomly that is roughly in the middle of the available values, to test a close-to-worst-jitter scenario, and with many bits switching to explore the whole characteristic of the DTC. The result is shown in Figure 5.10: the residual jitter remains within specification (35 ns_{rms} over the whole temperature range for all the samples tested but one.

A final test was performed by testing the IC+MEMS system together with the on-board temperature acquisition chain in a real live compensation of the temperature drift over a long interval of time of ≈20 h. The board was placed in an indoor uncontrolled environment and the on-board microcontroller was programmed to provide the correct CW to the chip to keep the output frequency stable. The result is shown in Figure 5.11. The frequency stays within the predicted ±10 ppm error range, even though it is expected that with an on-chip temperature sensor the output drift would have been much

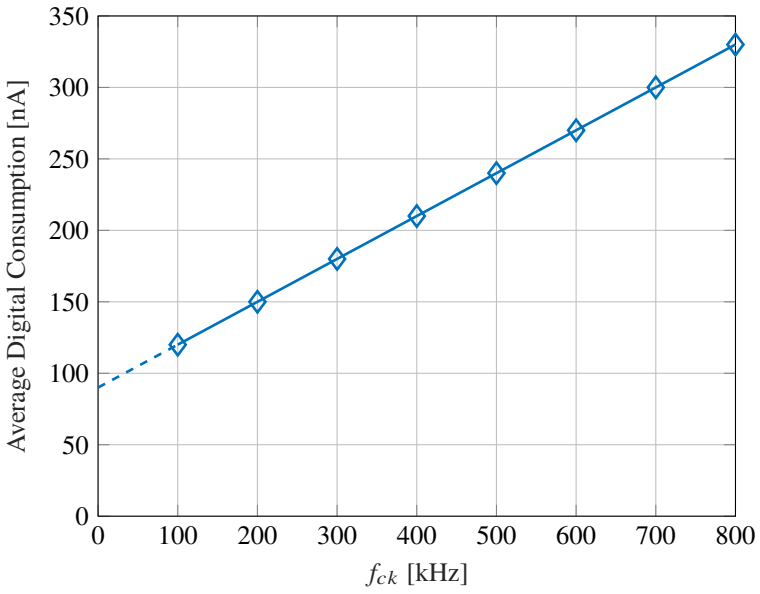


Figure 5.9: *Digital consumption measured as a function of the input clock frequency. It was not possible to measure a meaningful consumption at* $f_{ck} = 0\,\text{Hz}$ *because of the dynamic flip-flops, that introduce a static consumption. Therefore, from the interpolation of the curve it is possible ot find the static consumption due to the DLL Charge Pump and its bias branches, amounting to* 90 nA.

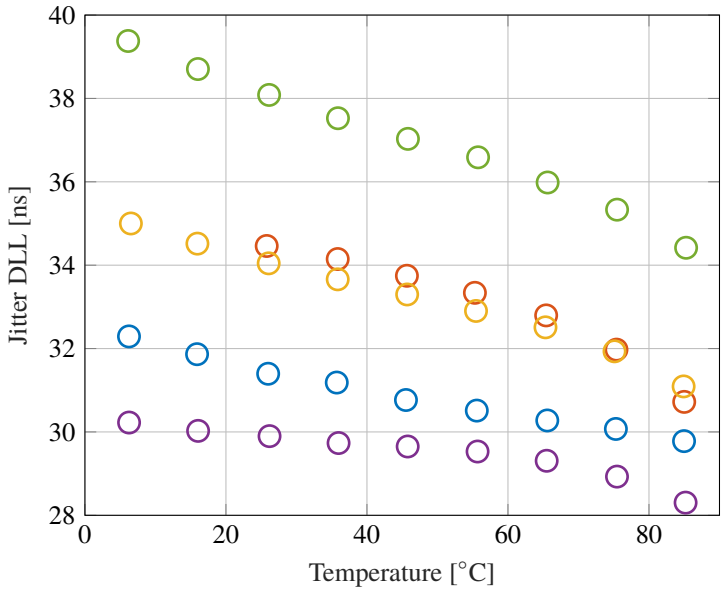


Figure 5.10: *Jitter of the digital blocks after the compensation over the temperature range. The drift of the residual jitter in temperature is due to the bias network that provides the current reference to the charge pump of the DLL, that was not optimized for stability over temperature.*

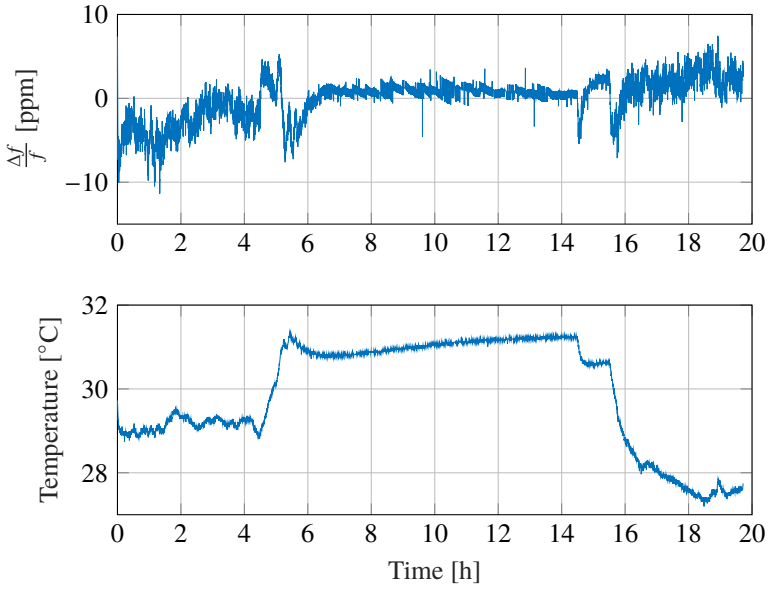


Figure 5.11: *Live compensation of the MEMS frequency in an uncontrolled environment over a long period of time. The biggest changes in the measured temperature are due to the switching on/off of the air conditioning system.*

smaller due to a better thermal homogeneity between the MEMS and the sensor. In this case, the clock would be expected to withstand temperature gradients as large as $1\,°\mathrm{C}\,\mathrm{s}^{-1}$.

Conclusions

An alternative approach towards MEMS-based low-power time-keeping devices has been proposed. The approach is alternative in that it builds on existing knowledge to study the limitations – but also the advantages! – of using a cheap, standard and with relatively poor thermal properties technological properties, a matter that had never been considered and analyzed in detail.

The hypothesis of a better repeatability of the thermal properties of the fabricated test structures was studied, proving that an uncalibrated part-to-part spread of ±5 ppm is achievable.

The implications of more severe thermal drift of the MEMS frequency at system-level were studied in detail: the most critical aspect was identified as the response to strong thermal gradients. This fact poses a strong limitation in the system sizings available to the designer with respect to RTCs based on crystalline MEMS resonators.

A jitter suppression scheme based on a Digital-to-Time Converter was proposed, which achieves a 160 nA average consumption, outperforming previous architectures based on a PLL phase filtering. This concept, together with a low-power oscillator and a fractional divider, was implemented on a silicon chip which, together with an on-board temperature sensor, enabled to demonstrate a functional RTC system.

Within the Thesis, the point of view of a big semiconductor company was often adopted and a particular emphasis was posed on the calibration costs that a RTC based on a MEMS produced in a custom, optimized process would meet. At present, only one company has been able to successfully

produce and sell in large amounts MEMS-based RTCs starting from a native MEMS drift of ±100 ppm across the range. This is a remarkable achievement; nevertheless, these parts sell nowadays for an amount in the order of $0.10 and the competition that is expected to arise in the field will be likely to cause a further reduction of the economic margins. The approach proposed that was proposed can be a successful strategy in this hostile market.

List of Figures

List of Tables

Bibliography

[1] G. Mussi, M. Bestetti, V. Zega, A. Frangi, and G. Langfelder, "An Outlook on Potentialities and Limits in Using Epitaxial Polysilicon for MEMS Real-Time Clocks", *IEEE Transactions on Industrial Electronics*, 2019. DOI: 10.1109/TIE.2019.2938465.

[2] G. Mussi, M. Bestetti, V. Zega, A. Frangi, G. Gattere, and G. Langfelder, "Resonators for real-time clocks based on epitaxial polysilicon process: A feasibility study on system-level compensation of temperature drifts", in *Proc. IEEE Micro Electro Mechanical Systems (MEMS)*, Jan. 2018, pp. 711–714. DOI: 10.1109/MEMSYS.2018.8346654.

[3] G. Mussi, P. Frigerio, G. Langfelder, and G. Gattere, "A Low-Power Deterministic Approach to Jitter Suppression in MEMS-Based Real-Time Clocks", in *Proc. Joint Conf. of the IEEE Int. Frequency Control Symp. and European Frequency and Time Forum (EFTF/IFCS)*, Apr. 2019, pp. 1–4. DOI: 10.1109/FCS.2019.8856038.

[4] G. Mussi, G. Langfelder, G. Gattere, and C. Valzasina, *Generatore di orologio in tempo reale a potenza ultra bassa utilizzante un risonatore microelettromeccanico e metodo di compensazione di jitter in tale generatore di orologio*, Italian, Patent Application IT2018000010577, 2019.

[5] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey", *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[6] S. Kamath and J. Lindh, "AN092 - Measuring Bluetooth Low Energy Power Consumption", Texas Instruments, Tech. Rep., 2012. [Online]. Available: http://www.ti.com/lit/an/swra347a/swra347a.pdf.

[7] T. Schmid, P. Dutta, and M. B. Srivastava, "High-resolution, Low-power Time Synchronization an Oxymoron No More", in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, ser. IPSN '10, Stockholm, Sweden: ACM, 2010, pp. 151–161, ISBN: 978-1-60558-988-6. DOI: 10.1145/1791212.1791231.

[8] F. Terraneo, F. Riccardi, and A. Leva, "Jitter-Compensated VHT and Its Application to WSN Clock Synchronization", in *Proc. IEEE Real-Time Systems Symp. (RTSS)*, Dec. 2017, pp. 277–286. DOI: 10.1109/RTSS.2017.00033.

[9] A. Partridge, H. C. Lee, P. Hagelin, and V. Menon, "We know that MEMS is replacing quartz. But why? And why now?", in *Proc. Joint European Frequency and Time Forum Int. Frequency Control Symp. (EFTF/IFCS)*, Jul. 2013, pp. 411–416. DOI: 10.1109/EFTF-IFC.2013.6702311.

[10] *SiTime still standing tall in MEMS timing devices industry*, [Online] https://archive.eetasia.com/www.eetasia.com/ART_8800720550_480100_NP_3390df06.HTM, Mar. 2016.

[11] M. Li, C. Chen, C. Li, C. Chin, and S. Li, "A Monolithic CMOS-MEMS Oscillator Based on an Ultra-Low-Power Ovenized Micromechanical Resonator", *Journal of Microelectromechanical Systems*, vol. 24, no. 2, pp. 360–372, Apr. 2015, ISSN: 1057-7157. DOI: 10.1109/JMEMS.2014.2331497.

[12] R. Melamud, B. Kim, S. A. Chandorkar, M. A. Hopcroft, M. Agarwal, C. M. Jha, and T. W. Kenny, "Temperature-compensated high-stability silicon resonators", *Applied physics letters*, vol. 90, no. 24, p. 244 107, 2007. DOI: 10.1063/1.2748092.

[13] J. J. Hall, "Electronic effects in the elastic constants of n-type silicon", *Physical Review*, vol. 161, no. 3, p. 756, 1967. DOI: 10.1103/PhysRev.161.756.

[14] E. J. Ng, V. A. Hong, Y. Yang, C. H. Ahn, C. L. Everhart, and T. W. Kenny, "Temperature dependence of the elastic constants of doped silicon", *Journal of Microelectromechanical Systems*, vol. 24, no. 3, pp. 730–741, 2014. DOI: 10.1109/JMEMS.2014.2347205.

[15] D. Ruffieux, F. Krummenacher, A. Pezous, and G. Spinola-Durante, "Silicon Resonator Based 3.2 μW Real Time Clock With ±10 ppm Frequency Accuracy", *IEEE Journal of Solid-State Circuits*, vol. 45, no. 1, pp. 224–234, Jan. 2010, ISSN: 0018-9200. DOI: 10.1109/JSSC.2009.2034434.

[16] D. Ruffieux, F. Pengg, N. Scolari, F. Giroud, D. Severac, T. Le, S. D. Piazza, and O. Aubry, "A 3.2 ×1.5 ×0.8mm3 240nA 1.25-to-5.5V 32kHz-DTCXO RTC module with an overall accuracy of μ1ppm and an all-digital 0.1ppm compensation-resolution scheme at 1Hz", in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Jan. 2016, pp. 208–209. DOI: 10.1109/ISSCC.2016.7417980.

[17] S. Zaliasl, J. C. Salvia, G. C. Hill, L. Chen, K. Joo, R. Palwai, N. Arumugam, M. Phadke, S. Mukherjee, H. C. Lee, C. Grosjean, P. M. Hagelin, S. Pamarti, T. S. Fiez, K. A. A. Makinwa, A. Partridge, and V. Menon, "A 3 ppm 1.5 × 0.8 mm 2 1.0 μA 32.768 kHz MEMS-Based Oscillator", *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 291–302, Jan. 2015, ISSN: 0018-9200. DOI: 10.1109/JSSC.2014.2360377.

[18] *Ultra-Small μPower, ±5 to ±20 ppm, 32.768 kHz TCXO*, SiTime Corp., 2014. [Online]. Available: https://www.sitime.com/datasheet/SiT1532.

[19] G. Langfelder, S. Dellea, F. Zaraga, D. Cucchi, and M. Azpeitia Urquia, "The Dependence of Fatigue in Microelectromechanical Systems on the Environment and the Industrial Packaging", *IEEE Transactions on Industrial Electronics*, vol. 59, no. 12, pp. 4938–4948, Dec. 2012, ISSN: 0278-0046. DOI: 10.1109/TIE.2011.2151824.

[20] G. Gattere, A. Tocchio, and C. Valzasina, *Microelectromechanical resonator with improved electrical features*, European Patent EP 3 407 492 A1, 2018.

[21] C. T.-. Nguyen and R. T. Howe, "An integrated CMOS micromechanical resonator high-Q oscillator", *IEEE Journal of Solid-State Circuits*, vol. 34, no. 4, pp. 440–455, Apr. 1999, ISSN: 0018-9200. DOI: 10.1109/4.753677.

[22] E. A. Vittoz, M. G. Degrauwe, and S. Bitz, "High-performance crystal oscillator circuits: Theory and application", *IEEE Journal of Solid-State Circuits*, vol. 23, no. 3, pp. 774–783, 1988. DOI: 10.1109/4.318.

[23] A. Frangi and P. Guerrieri, "Modelling geometrical nonlinearities in MEMS resonators", in *First International Nonlinear Dynamics Conference(Nodycon)*, 2019.

[24] T. Amemiya, *Advanced econometrics*. Harvard university press, 1985.

[25] K. A. A. Makinwa, *Smart Temperature Sensor Survey*, Jul. 2019. [Online]. Available: http://ei.ewi.tudelft.nl/docs/TSensor_survey.xls.

[26] B. Yousefzadeh, S. Heidary Shalmany, and K. A. A. Makinwa, "A BJT-Based Temperature-to-Digital Converter With ±60 mK (3 σ) Inaccuracy From -55 °C to +125 °C in 0.16-μm CMOS", *IEEE Journal of Solid-State Circuits*, vol. 52, no. 4, pp. 1044–1052, Apr. 2017, ISSN: 0018-9200. DOI: 10.1109/JSSC.2016.2638464.

[27] T. Zhang, M. Law, P. Mak, M. Vai, and R. P. Martins, "Nano-Watt Class Energy-Efficient Capacitive Sensor Interface With On-Chip Temperature Drift Compensation", *IEEE Sensors Journal*, vol. 18, no. 7, pp. 2870–2882, Apr. 2018. DOI: 10.1109/JSEN.2018.2803203.

[28] A. Lahiri, N. Gupta, A. Kumar, and P. Dhadda, "A 600 µA 32 kHz Input 960 MHz Output CP-PLL With 530 ps Integrated Jitter in 28 nm FD-SOI Process", *IEEE Journal of Solid-State Circuits*, vol. 50, no. 7, pp. 1680–1689, Jul. 2015. DOI: 10.1109/JSSC.2015.2412680.

[29] S. E. Meninger and M. H. Perrott, "A 1-MHZ bandwidth 3.6-GHz 0.18-μm CMOS fractional-N synthesizer utilizing a hybrid PFD/DAC structure for reduced broadband phase noise", *IEEE Journal of Solid-State Circuits*, vol. 41, no. 4, pp. 966–980, Apr. 2006. DOI: 10.1109/JSSC.2006.870894.

[30] E. Temporiti, G. Albasini, I. Bietti, R. Castello, and M. Colombo, "A 700-kHz bandwidth $\Sigma\Delta$ fractional synthesizer with spurs compensation and linearization techniques for WCDMA applications", *IEEE Journal of Solid-State Circuits*, vol. 39, no. 9, pp. 1446–1454, Sep. 2004. DOI: 10.1109/JSSC.2004.831598.

[31] A. L. Lacaita, S. Levantino, and C. Samori, *Integrated frequency synthesizers for wireless systems*. Cambridge University Press, 2007.

[32] D. Ruffieux, "A high-stability, ultra-low-power quartz differential oscillator circuit for demanding radio applications", in *Proc. 28th European Solid-State Circuits Conf*, Sep. 2002, pp. 85–88.

[33] D. Ruffieux, *Differential oscillator circuit including an electro-mechanical resonator*, US Patent 6,956,443, 2005.

[34] G. W. Pierce, "Piezoelectric crystal resonators and crystal oscillators applied to the precision calibration of wavemeters", in *Proceedings of the American Academy of Arts and Sciences*, JSTOR, vol. 59, 1923, pp. 81–106. DOI: 10.2307/20026061.

[35] S. Levantino, M. Zanuso, C. Samori, and A. Lacaita, "Suppression of flicker noise upconversion in a 65nm CMOS VCO in the 3.0-to-3.6GHz band", in *Proc. IEEE Int. Solid-State Circuits Conf. - (ISSCC)*, Feb. 2010, pp. 50–51. DOI: 10.1109/ISSCC.2010.5434054.

[36] B. Razavi and R. Behzad, *RF microelectronics*. Prentice hall New Jersey, 1998, vol. 1.

[37] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolic, *Digital integrated circuits*. Prentice hall Englewood Cliffs, 2002, vol. 2.

[38] D. Harris and N Weste, "CMOS VLSI Design", *Pearson Education, Inc*, 2010.

[39] S. Pellerano, S. Levantino, C. Samori, and A. L. Lacaita, "A 13.5-mW 5-GHz frequency synthesizer with dynamic-logic frequency divider", *IEEE Journal of Solid-State Circuits*, vol. 39, no. 2, pp. 378–383, Feb. 2004. DOI: 10.1109/JSSC.2003.821784.

[40] G. W. Roberts and M. Ali-Bakhshian, "A Brief Introduction to Time-to-Digital and Digital-to-Time Converters", *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 3, pp. 153–157, Mar. 2010, ISSN: 1549-7747. DOI: 10.1109/TCSII.2010.2043382.

[41] D. K. Jeong, G. Borriello, D. A. Hodges, and R. H. Katz, "Design of PLL-based clock generation circuits", *IEEE Journal of Solid-State Circuits*, vol. 22, no. 2, pp. 255–261, Apr. 1987. DOI: 10.1109/JSSC.1987.1052710.

[42] J. G. Maneatis, "Low-jitter process-independent DLL and PLL based on self-biased techniques", *IEEE Journal of Solid-State Circuits*, vol. 31, no. 11, pp. 1723–1732, Nov. 1996, ISSN: 0018-9200. DOI: 10.1109/JSSC.1996.542317.

[43] C. S. Taillefer and G. W. Roberts, "Delta–Sigma A/D Conversion Via Time-Mode Signal Processing", *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 9, pp. 1908–1920, Sep. 2009. DOI: 10.1109/TCSI.2008.2010144.

[44] S. Sidiropoulos and M. A. Horowitz, "A semidigital dual delay-locked loop", *IEEE Journal of Solid-State Circuits*, vol. 32, no. 11, pp. 1683–1692, Nov. 1997, ISSN: 0018-9200. DOI: 10.1109/4.641688.

[45] C. Sharpe, "3-STATE PHASE DETECTOR CAN IMPROVE YOUR NEXT PLL DESIGN.", *EDN*, vol. 21, no. 17, pp. 55–59, 1976.

[46] C.-K. Ken Yang, *Delay-locked loops - An overview*. 2003, pp. 13–22. DOI: 10.1109/9780470545492.ch2.

[47] E. Hegazi and A. A. Abidi, "A 17-mW transmitter and frequency synthesizer for 900-MHz GSM fully integrated in 0.35-μm CMOS", *IEEE Journal of Solid-State Circuits*, vol. 38, no. 5, pp. 782–792, May 2003, ISSN: 0018-9200. DOI: 10.1109/JSSC.2003.810052.

[48] A. A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators", *IEEE Journal of Solid-State Circuits*, vol. 41, no. 8, pp. 1803–1816, Aug. 2006, ISSN: 0018-9200. DOI: 10.1109/JSSC.2006.876206.