UNIVERSITÀ DEGLI STUDI DI PAVIA
FACOLTA' DI INGEGNERIA
DIPARTIMENTO DI ELETTRONICA

DOTTORATO DI RICERCA IN MICROELETTRONICA
XXIII CICLO

DIGITAL PHASE LOCKED LOOPS FOR WIRELESS COMMUNICATIONS

TUTORE:
CHIAR.MO PROF. RINALDO CASTELLO

COORDINATORE:
CHIAR.MO PROF. FRANCO MALOBERTI

TESI DI DOTTORATO DI
LUCA VERCESI

Contents

Introduction

In past decades, "Moore's law" has governed the evolution in microelectronics. Through continuous advancements in device and fabrication technology, the industry has maintained exponential progress rates in transistor miniaturization and integration density. As a result, microchips have become cheaper, faster, more complex and more power efficient.

A reduced feature size combined with a lower supply voltage are the key technology drivers resulting in reduced cost, higher speed, and lower power consumption of digital integrated circuits. For analog circuits, scaling is less attractive. Unlike digital circuits, analog functions are constrained by electronic noise and accuracy requirements, factors that only conditionally benefit from technology scaling and that can even deteriorate for very low supply voltages.

Although digital circuits greatly benefit from the technology evolution, analog circuits are still needed in most applications. Applications such as audio, video, and RF (radio frequency) communications require analog circuits as interface with the physical world. The design of highly integrated mixed signal systems become more challenging as the relative performance gap between analog and digital circuits widens. These systems are increasingly limited not by the available digital processing power, but by their analog interfaces.

Digitally assisted analog circuits have been emerging in the last decades as a design approach to the mixed signal systems. The idea is that analog circuits performances can be enhanced by digital logic. For example extended calibration and post-processing algorithms can be designed exploiting the growing computational power of Digital Signal Processing (DSP). The digitally assisted analog circuit design was initially applied to Analog to Digital Converters (ADC), that are the basic elements of mixed signal circuits. Some examples of this design approach are gain calibration techniques for pipelined

ADC, dynamic element matching in Digital to Analog Converters (DAC) and various foreground and background calibration algorithms [1]. In mixed signal systems the digital portion is therefore getting wider, while the analog circuits are relegated to boundary of the system. However, even if the analog circuits are less extensively used, they still represent a key element in the system performances an their design is getting ever more challenging.

The Phase Locked Loop (PLL) is a control system that is capable to generate stable and finely programmable frequencies. It is an important element of a great variety of systems because of its remarkable versatility. The PLL fields of application are frequency syntheses, clock distribution, and clock and data recovery. PLLs input and output signals are intrinsically analog; and therefore PLLs are commonly designed as purely analog systems. As the technology evolved to more efficient digital circuits, the idea of digital PLL has emerged (despite of their name, digital PLLs are mixed signal systems). A number of works on digital PLLs have been published in the last decade ([2, 10, 15, 16, 18, 21]), and they shows different approaches to the system design.

The target of the activity described in this dissertation is the design of a digitally intensive PLL. The digitally assisted analog circuit design methodology has been widely applied, extensive digital logic is used to relax analog circuits specifications. Thanks to the collaboration with the Italian design center of Marvell, the research activity was oriented to real application an tightly linked to the industrial context.

The activity was organized in two main steps: the design of a Time to Digital Converter (TDC), and the system level design of an All Digital Phase Locked Loop (ADPLL). TDC is a particular type of Analog to Digital Converter and is a key block in the ADPLL. A TDC was designed, from the topology down to the circuit level (including full custom layout), resulting in a prototype that was realized and characterized. The ADPLL design was started from system level. Various calibration and acquisition aiding algorithms were studied, that was developed form the behavioral to the digital implementation design. A prototype of the ADPLL was realized, its characterization is still in progress.

The dissertation starts with a brief introduction to some basic concepts concerning Phase Locked Loops (**chapter 1**). In **chapter 2** the ADPLL is pre-

sented, system level considerations are developed highlighting the impact and the limitations of major building blocks. The Time to Digital Converter and the Digitally Controlled Oscillator are discussed respectively in **chapters 3** and **4**, together with some related calibration algorithms. **Chapter 5** is dedicated to the ADPLL phase locking acquisition; ADPLL self locking limits are presented together with acquisition aiding algorithms. The TDC and the ADPLL prototypes are discussed (including measurement results) in **chapter 6**.

Chapter 1

The Phase Locked Loop

Contents

In this chapter the Phase Locked Loop (PLL) is introduced. The purpose of this chapter is to brush up some basic PLL concepts that will be recalled in the dissertation. For a detailed and extended description of the concepts that are presented in this chapter the reader can refers to dedicated publications and textbooks ([3, 4, 5]).

1.1 PLL overview

The Phase Locked Loops are basically feedback control systems composed of a Phase Detector (PD), a Loop Filter and a Local Oscillator (LO). The LO frequency has to be tunable; usually a Voltage Controlled Oscillator (VCO) is used as LO. In a VCO the oscillation frequency is controlled by a voltage signal. A reference oscillator generates the PLL input, that is compared (by the PD) with the locally generated signal to produce a signal proportional to

their instantaneous phase difference (ϕ_e). The PD output is filtered and used to control the instantaneous frequency of the LO. Therefore the PLL controls the LO frequency, that tracks the reference frequency.

A frequency divider is usually added between LO and PD for frequency synthesis applications. A block diagram of the PLL is shown in Fig. 1.1. The presence of the frequency divider in the feedback path allows the LO frequency to be a multiple of the reference frequency. Even if the reference and LO frequency are different, the LO still tracks the reference; this means that two frequency are in a fixed ratio.

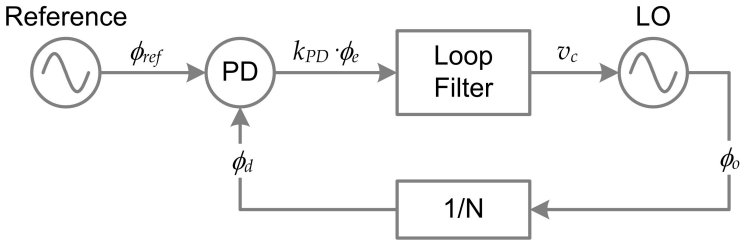

Figure 1.1: Phase locked loop and phase domain model variables

1.2 PLL linear model

Although PLLs are inherently nonlinear circuits, their behavior can be approximated very well by linear models. A linear model typically is applicable if phase error is small, a condition normally attained when the loop is locked. Most of the analysis and design of PLLs can be based on the linear approximations; analysis becomes far more challenging when the linear approximations fail.

Among the tools of linear analysis, Laplace and Fourier transforms and various concepts derived therefrom stand out as being particularly valuable. The related concept of a transfer function, describing a transform-domain relation between input and output of a linear circuit, is an extremely powerful tool for dealing with PLLs. Analytical design of PLLs is carried out almost entirely

through transfer functions.

Furthermore the most pervasive classification of PLLs (both analog and digital) relies on open-loop transfer function properties. This classification is based on the **order** and the **type** of the PLL; the meaning of this parameters will be treated in the next pages.

In ordinary electrical circuits, a transfer function relates voltages and currents of input and output signals. But in a PLL, the main variable is the phase; therefore the PLL will be represented in a phase domain linear model. Several textbooks treat this topic ([3, 5]).

1.2.1 Linear model building blocks

Let's consider the system represented in figure 1.1. The phase of the reference oscillator is denoted as ϕ_{ref} and the phase of the LO and of the divider outputs are denoted respectively ϕ_o and ϕ_d. Assume that the loop is locked and that the PD is linear so that the PD output signal is

$$v_{PD} = k_{PD} \left(\phi_{ref} - \phi_d \right) \tag{1.1}$$

where k_{PD} is the phase detector gain and is measured in units of volt[1] per radian. Define the phase error as

$$\phi_e = \phi_{ref} - \phi_d \tag{1.2}$$

The error signal v_{PD} is processed by the loop filter, which defines the dynamic performance of the system. The transfer function of the filter is denoted as $F(s)$. The loop filter output (v_c) is the control signal of the LO. In the Laplace transform domain[2], the action of the filter is described by;

$$V_c(s) = F(s) \cdot V_{PD}(s) \tag{1.3}$$

The LO frequency can be assumed proportional to v_c and therefore the LO output phase can be expressed in the Laplace domain as

$$\phi_o(s) = \frac{k_{LO} \cdot V_c(s)}{s} \tag{1.4}$$

[1] If the output signal is expressed in volts, otherwise this unit change as applicable.

[2] $V_c(s)$ is the Laplace transform of $v_c(t)$, similar notation is used for other variables; generally the time or Laplace domain is indicated by the t or s in the parenthesis.

were k_{LO} is the gain of the LO, that is expressed in units of Hertz per volt[1]. This equation directly comes from the intrinsic integrative relationship between phase and frequency. The divider output phase is expressed as

$$\phi_d = \frac{\phi_o}{N} \tag{1.5}$$

were N is the frequency divider factor.

Since equations 1.1-1.5 are valid in almost every PLL, the properties of the PLL are defined by the loop filter. The loop filter characteristic equation may assume different aspects due to the specific design constraints.

1.2.2 PLL transfer functions

PLL system's transfer functions can be derived combining the characteristic equations of the building blocks. This functions are useful to the design process and will be recalled throughout the others chapters.

Applying the feedback theory the open loop transfer function $G(s)$ can be used to derive all the others transfer functions. The following equations apply to all PLL having the configuration of figure 1.1, irrespective of the loop filter properties.

- Open loop transfer function

$$G(s) = \frac{k_{PD} \cdot k_{LO} \cdot F(s)}{N \cdot s} \tag{1.6}$$

- System transfer function:

$$H(s) = \frac{\phi_o(s)}{\phi_{ref}(s)} = \frac{N \cdot G(s)}{1 + G(s)} = \frac{N \cdot k_{PD} \cdot k_{LO} \cdot F(s)}{N \cdot s + k_{PD} \cdot k_{LO} \cdot F(s)} \tag{1.7}$$

- Error transfer function:

$$E(s) = \frac{\phi_e(s)}{\phi_{ref}(s)} = \frac{1}{1 + G(s)} = \frac{N \cdot s}{N \cdot s + k_{PD} \cdot k_{LO} \cdot F(s)} \tag{1.8}$$

The open loop transfer function gain is generally a monotonically descending function of frequency with at least one pole at zero frequency. The $G(S)$

unitary gain frequency is the PLL **bandwidth**. The system transfer function is a low-pass function with DC gain equal to N. The cutoff frequency of the system transfer function is the PLL bandwidth. The error transfer function is a high-pass function with the corner frequency corresponding to the PLL bandwidth.

The dynamic properties of the PLL (for example the impulse and step response to a phase of a frequency perturbation) directly depends on the system transfer functions. The PLL **order** is the number of poles in the system transfer function. However, many important and general properties of a loop relate to the PLL **type**, that is the number of integrators (poles at zero frequency) in the open loop transfer function.

The most common PLL type is the type two; however depending on the specific field of application and of the design constraints different PLL type and order may be chosen. The type two (and higher types) PLL has a zero average phase error, while in type one the average phase error depends both on the PLL operating conditions and on the loop filter design. The zero average phase error is a system requirement in many different cases, and is the principal reason for the widespread use of type two PLLs. PLL orders higher than two are rarely used; type three PLLs are used in specific telecommunication systems where the Doppler effect can produce significant frequency ramps ([3]).

1.2.3 Noise in PLLs

Different metrics exist to quantify the noise performance of an oscillator. Jitter is used to quantify the noise effects that are observed in the time domain, while the Phase Noise (PN) is used in the frequency domain.

Since the oscillators are not Linear Time Invariant (LTI) systems, all the LTI theoretical analysis tools does not apply; therefore theoretical approaches to timing Jitter and Phase Noise have been proposed ([6, 7, 8]), that involve more complicated maths.

At the system level it is still possible to get some insight of the noise effects using the linear model transfer function. Two noise sources are of particular interest: the PD and the LO. In figure 1.2 the PLL linear model is represented with the PD and LO phase noise sources.

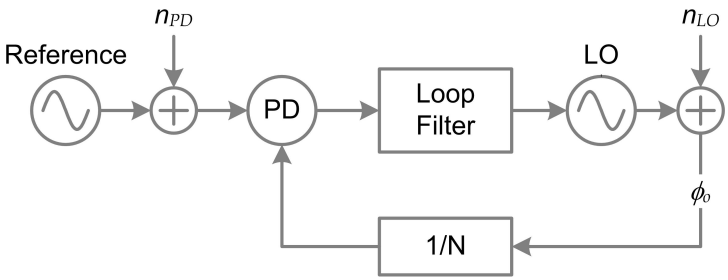

Figure 1.2: Phase locked loop noise sources

The transfer function between the PD noise source and the LO output phase is the system transfer function:

$$\frac{\phi_o(s)}{n_{PD}(s)} = H(s) = \frac{\phi_o(s)}{\phi_{ref}(s)} = \frac{N \cdot k_{PD} \cdot k_{LO} \cdot F(s)}{N \cdot s + k_{PD} \cdot k_{LO} \cdot F(s)} \tag{1.9}$$

As introduced before this is a low pass transfer function with a cutoff frequency corresponding to the PLL bandwidth.

The transfer function between n_{LO} and ϕ_o is equal to the error transfer function

$$\frac{\phi_o(s)}{n_{LO}(s)} = E(s) = \frac{\phi_e(s)}{\phi_{ref}(s)} = \frac{N \cdot s}{N \cdot s + k_{PD} \cdot k_{LO} \cdot F(s)} \tag{1.10}$$

that is a high pass function with the corner frequency equal to PLL bandwidth and unitary constant gain at high frequency.

Acting on the PLL bandwidth it is not possible to reduce both the impact of the PD and LO noise sources. This results in a design trade-off between wide-band PLL that reduce the impact of LO noise sources, and narrow-band PLL that reduce the impact of PD noise sources.

The PLL is also affected by deterministic disturbances that may be produced by the divider or the PD itself. This disturbances are related to fractional spurs, and will be discussed in the section 2.3. Also this deterministic disturbances have a low-pass transfer function towards the LO output phase.

The combined presence of PD noise sources and deterministic disturbances usually make narrow band PLL design more attractive.

1.3 Acquisition of phaselock

The use of the linear phase domain model is based on the assumption that the loop is in lock. However when the PLL is activated and the loop is closed, the system starts out in an unlocked condition and must be brought into lock. The process of bringing a loop into lock is called **acquisition** ([3]).

The acquisition process has to be studied considering the large signal effects. Acquisition is inherently a nonlinear phenomenon; nonlinear analysis is needed generally, without easy help from linear approximations.

A type n PLL contains n integrators. Each integrator is associated to a state variable of the system: phase, frequency, frequency rate, and so on. To bring the loop into lock, it is necessary to set each of the state variables into close agreement with the corresponding conditions of the input signal. Therefore, it is necessary to plan for phase acquisition, frequency acquisition, and so on, up to n forms of acquisition for a type n loop. The higher the type of the system, the more the design of the PLL is complicated by the request of robust acquisition.

Telecommunication standard put some constraints on the acquisition process. High reliability is required for the acquisition process, and a fast locking is generally appreciated. When the system self-acquisition performance are unacceptably poor, acquisition aiding circuits are usually added to the system. This point will be addressed in the chapter 4.

1.4 Frequency Locked Loops

The FLL (figure 1.3) is a feedback control systems constituted by a Frequency Detector (FD), a Loop Filter and a Local Oscillator (LO).

A Frequency Locked Loop (FLL) is a system very similar to the PLL. The two system relies on the detection of frequency and phase respectively, that are deeply related entities.

$$\phi(t) = \int_{t_0}^{t} f(t)\, dt \tag{1.11}$$

The FLL can be analyzed as the PLL through a linear phase domain model.

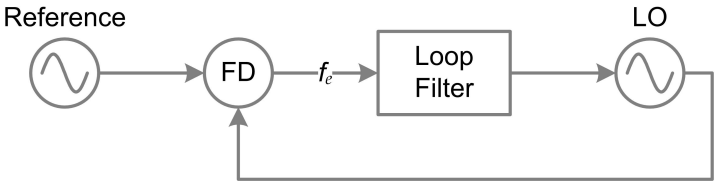

Figure 1.3: Frequency Locked Loop

In this case the FD exhibits a derivative behavior due to the transformation between the phase output of the LO and the FD frequency error output signal.

FLL are not suitable for application that demands the absolute phase of the LO to be under control. FLL design is simpler than PLL design, because of the presence of a single state variable in the system (the output frequency). In particular the presence of a single state variable in the system make the acquisition process of an FLL much simpler than that of a PLL.

An FLL is usually added to PLL system as auxiliary aquisition aiding circuit. More often the PD is replaced by a Phase Frequency Detector (PFD). A PFD ([3, 5]) combine the small signal behavior of a phase detector, with the large signal behavior of a frequency detector. The use of PFD is very popular in the design of analog PLLs, but it will not longer be recalled in this dissertation.

Chapter 2

Digital intensive frequency synthesis

Contents

The continued progress in increasing performance, speed, reliability, and the simultaneous reduction in size and cost of digital integrated circuits has resulted in strong interest in the implementation of the PLL in the digital domain.

A number of works on digital PLLs have been published since 1960 [2]; the earliest efforts on Digital PLLs were focused on partially replacing the analog PLL components with digital ones. The All Digital PLL (ADPLL) is the result of this evolution process.

2.1 ADPLL system overview

An ADPLL is characterized by the absence of analog signals other than the LO Radio Frequency (RF) signal, the Reference signal and the frequency divider output. While the analog nature of the LO RF signal is clear, the analog nature of the typically squared wave Reference and the divider output signals may be in doubt. The analog nature of all this signals is inherent in their timing and thus in their phase.

A block diagram of an ADPLL is shown in Fig. 2.1. A Time to Digital Converter (TDC) digitalizes the delay between the divider output and the reference signal replacing the analog PD. The loop filter is digitally implemented in the Z domain generating the Frequency Control Word (FCW), which drives the Digitally Controlled Oscillator (DCO) [1] .

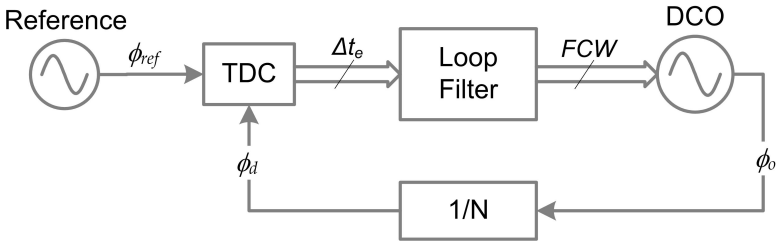

Figure 2.1: All Digital Phase locked loop

This block diagram applies to almost every ADPLL, even if this scheme is usually augmented by other circuits that may be required for phase and frequency modulation, phase locking acquisition or for calibration procedures.

[1] In an ADPLL the LO is implemented by a DCO. Both the names "LO" and "DCO" are used in this dissertation, within the ADPLL system they refer to the same block. The term "LO" is used when the nature of the control signal is not relevant, while the term "DCO" is used to emphasize the quantized control word

2.1.1 TDC quantization impact on the ADPLL

Consider an ideal TDC which produces an output code that is directly proportional to the delay between the input signals[2]. Assuming the locking condition for the ADPLL , the TDC output Δt_e in the linear model presented in the chapter 1 is expressed as:

$$\Delta t_e = k_{TDC} \frac{\phi_{ref} - \phi_d}{\omega_{ref}} \tag{2.1}$$

were ϕ_{ref} is the reference signal phase, ϕ_d is the divider output phase and ω_{ref} is the reference signal angular frequency. The variable k_{TDC} represents the TDC gain, this parameter is prone to Process Voltages and Temperature (PVT) variations.

Every ADC is characterized by a finite resolution, this parameter in the TDC is represented by a time resolution Δt_{res}. In closed-loop operation, the TDC quantization affects the phase noise at the ADPLL output ([9]). Under a large-signal assumption (which means that the TDC output signal spans multiple quantization levels), the variance of the timing uncertainty added by the quantization is

$$\sigma_t^2 = \frac{(\Delta t_{res})^2}{12} \tag{2.2}$$

The phase noise (expressed in radians) is obtained by normalizing the standard deviation of the timing error to the Reference period and multiplying it by 2π radians:

$$\begin{aligned} \sigma_\phi &= 2\pi \frac{\sigma_t}{T_{ref}} \\ T_{ref} &= \frac{1}{f_{ref}} \end{aligned} \tag{2.3}$$

were T_{ref} is the period of the reference signal. The total phase noise power is spread uniformly over the span from zero to the Nyquist frequency [3].

[2] A extensive description of the TDC behavior will be treated in Chapter 3

[3] The Nyquist frequency is half the sampling frequency f_{ref}.

The single-sided spectral density is, therefore, expressed as

$$\mathcal{L}(\Delta f) = \frac{\sigma_\phi^2}{f_{ref}} \tag{2.4}$$

Consequently, the phase noise associated to the TDC timing quantization is

$$\mathcal{L}(\Delta f) = \frac{(2\pi)^2}{12}\left(\frac{\Delta t_{res}}{T_{ref}}\right)^2 \frac{1}{f_{ref}} \tag{2.5}$$

In chapter 1 a phase domain linear model was introduced. The quantization noise of the TDC is added to the system in the same way of ϕ_{ref} (section 1.2.3, equation 1.9). The transfer function between the TDC noise and the LO output is, therefore, the system transfer function $H(s)$ defined in equation 1.7. The system transfer function is a low pass transfer function with gain equal to the ratio between f_{LO} and f_{ref} (N).

2.1.2 DCO quantization impact on the ADPLL

The DCO is a key element of the ADPLL. It can be modeled as the cascade of a quantizer (a non linear block) and a VCO. The DCO output phase (ϕ_{DCO}) is therefore expressed in the time domain as

$$\phi_{DCO}(t) = 2\pi \int_{t_0}^{t} q(FCW(t))\, dt \tag{2.6}$$

were $q(x)$ is the quantizer nonlinear function. The quantizer resolution Δf_{res} corresponds to the DCO frequency resolution.

Following the same procedure that was used to derive the effect of the TDC quantization (section 2.1.1), it is possible to evaluate the impact of the DCO quantization ([9]).

Assuming that the FCW spans multiple quantization levels, the DCO frequency quantization error can be modeled as an additive uniformly distributed random variable with white noise spectral characteristics. The corresponding quantization error variance is

$$\sigma_f^2 = \frac{(\Delta f_{res})^2}{12} \tag{2.7}$$

The total frequency noise power is spread uniformly from zero to the Nyquist frequency. The single-sided spectral density is, therefore, expressed as

$$\frac{1}{2}S_f = \frac{\sigma_f^2}{f_{ref}} \tag{2.8}$$

The ideal integrator due to the conversion from frequency control to output phase (equation 2.6) can be expressed in the Laplace domain as

$$H_{int}(s) = \frac{2\pi}{s} \tag{2.9}$$

The resulting phase noise spectrum at the DCO output due to the DCO frequency quantization is

$$\mathcal{L}(\Delta f) = \frac{1}{12}\left(\frac{\Delta f_{res}}{\Delta f}\right)^2 \frac{1}{T_{ref}} \tag{2.10}$$

This equation is valid from zero to Nyquist frequency. To describe the phase noise at frequencies higher than Nyquist it is necessary to consider the effect of holding the FCW value between two different samples. Equation 2.10 has to be multiplied by the sinc function corresponding to the Fourier transform of the zero-order hold operation:

$$\mathcal{L}(\Delta f) = \frac{1}{12}\left(\frac{\Delta f_{res}}{\Delta f}\right)^2 \frac{1}{T_{ref}} sinc^2\left(\frac{\Delta f}{f_{ref}}\right) \tag{2.11}$$

The effect of the loop on this additive noise can be evaluated through the transfer function described by equation 1.10 (section 1.2.3). This is a high pass function with the corner frequency equal to PLL bandwidth and unitary magnitude at high frequency.

2.2 Fractional-N ADPLL

The fundamental requirement of a frequency synthesizer is the capability to synthesize any frequency (within a given range) independently from the ratio between the synthesized and the reference frequency. The designed system is

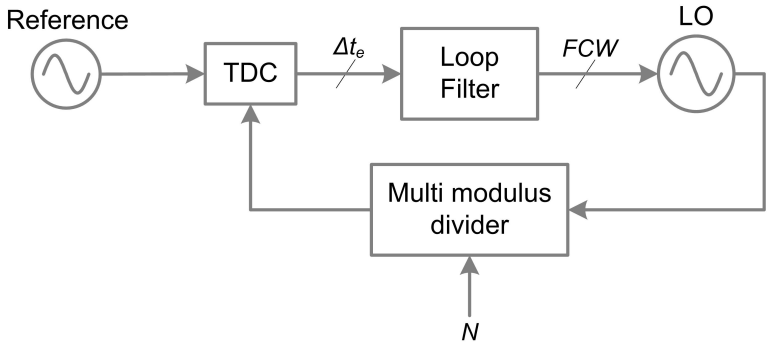

Figure 2.2: Fractional-N ADPLL

based on a standard Fractional-N architecture. This architecture (Fig. 2.2) is based on a multi modulus divider placed in the feedback path. The divider is controlled by an integer sequence N_k whose average value n is fractional. When the phase locking between the divider output and the reference signal[4] is achieved, the LO average frequency is:

$$\langle f_{LO} \rangle = n \cdot f_{ref} \tag{2.12}$$

Divider-less ADPLLs have been demonstrated ([9] and [10]); their main claim is that those structures are theoretically not affected by fractional spurs (this point will be addressed in section 2.3). Unfortunately those solutions have some drawbacks compared to the traditional multi modulus divider architecture. The main disadvantage is that both the counter and the TDC are driven by the LO RF output (Fig. 2.3) affecting the power consumption of the TDC and complicating its design. Furthermore the synchronization of TDC's and counter's outputs is an issue that has been addressed ([10]) but this problem simply disappears when the TDC and the multi modulus divider are connected in series. The multi modulus divider and the counter perform very similar tasks in the respective systems. This behavioral resemblance appears

[4]The reference frequency is assumed to be a constant value.

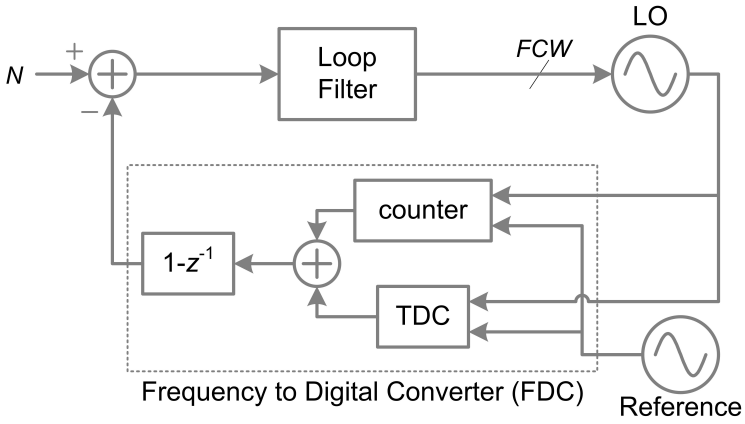

Figure 2.3: Divider less ADPLL

also in similar area occupancy and power consumption. Therefore the absence of the multi modulus divider results in minor advantages in terms of area and power.

2.3 Fractional spurs

Fractional-N synthesis suffer from a critical drawback: "fractional spurs". To understand the issue, consider the Fractional-N ADPLL depicted in Fig. 2.2, assuming open loop and a constant LO frequency that is $3 + 1/4$ times the reference frequency. As shown in Fig. 2.4, each of the first three cycles of the divided signal is slightly shorter than the reference period ($N = 3$). Consequently, the delay between the reference and the divider output (equation 2.13) grows in every reference period, until it returns to zero when divide-by-4 occurs.

$$\begin{aligned} \Delta t_k &= \Delta t_{k-1} + (T_{ref} - N_k \cdot T_{LO}) \\ &= \Delta t_{k-1} + (n - N_k) \cdot T_{LO} \end{aligned} \tag{2.13}$$

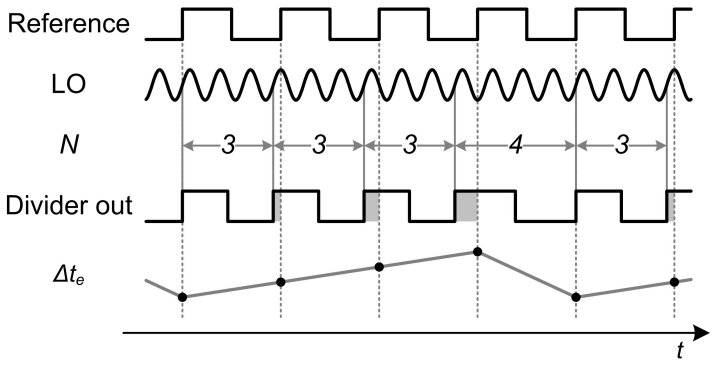

Figure 2.4: Fractional-N time diagram

If the LO frequency has to be equal to a fractional multiple of f_{ref}, then the TDC output signal (Δt_e) will be characterized by a repetitive ramp waveform. When the ADPLL loop is closed, such a waveform would modulate the LO, creating sidebands which are called fractional spurs. Fractional spurs are a critical drawback of the fractional-N architecture and they have to be seriously taken in account in an ADPLL design. Wide band systems are more affected by fractional spurs, since spurs are less filtered.

2.3.1 Fractional-N PLL with first order $\Sigma\Delta$ modulator

The example of figure 2.4 shows the effects of a particular sequence of N_k. Such a N_k sequence has been obtained as:

$$N_k = \lfloor n \rfloor + \Sigma\Delta_k \left(n - \lfloor n \rfloor \right) \tag{2.14}$$

were $\lfloor n \rfloor$ is the lower integer of n, and $\Sigma\Delta_k \left(x \right)$ is the k^{th} output of a first order Sigma Delta modulator with input x. An exhaustive theoretical approach to first order $\Sigma\Delta$ modulators (figure 2.5) can be found in [11]. From figure 2.5, the first order $\Sigma\Delta$ modulator system can be described by the differential

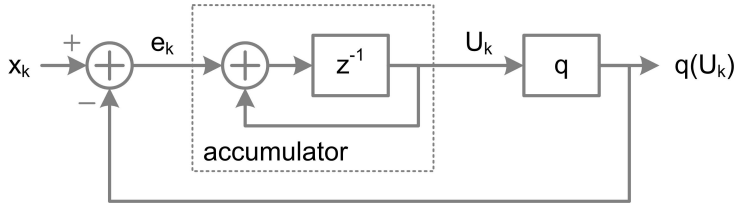

Figure 2.5: First order $\Sigma\Delta$ modulator

equations

$$\begin{aligned} U_k &= e_{k-1} + U_{k-1} \\ e_k &= x_k - q(U_k) \\ k &\in \mathbb{Z}^+ \end{aligned} \tag{2.15}$$

with initial condition

$$U_0 = u_0$$

where the binary quantizer is defined by

$$q(u) = \begin{cases} 1 & \text{if } u \geq \frac{1}{2} \\ 0 & \text{if } u < \frac{1}{2} \end{cases} \tag{2.16}$$

When the $\Sigma\Delta$ modulator input is a constant value x it is possible to demonstrate ($Corollary\ 1$ of [11]) that

$$\left| x - \frac{1}{K} \sum_{i=0}^{K-1} q(U_i) \right| \leq \frac{1}{K} \tag{2.17}$$

this theorem states that the more modulator output samples are averaged, the more the average is an accurate estimation of the input. This theorem highlight that the average of the divider control sequence define in equation 2.14 is equal to n as it is required.

Recalling the expression of the delay between reference and divider output (eq. 2.13), it is possible to relate the delay Δt_k to the $\Sigma\Delta$ properties. Manipulating the equations 2.13 and 2.14, we obtain:

$$\begin{aligned} \frac{\Delta t_k}{T_{LO}} &= \frac{\Delta t_{k-1}}{T_{LO}} + (n - N_k) \\ &= \frac{\Delta t_{k-1}}{T_{LO}} + [(n - \lfloor n \rfloor) - \Sigma\Delta_k (n - \lfloor n \rfloor)] \end{aligned} \tag{2.18}$$

It is possible to relate the variables of equation 2.18 to the input and the output variables of the $\Sigma\Delta$ model

$$\begin{aligned} x_k &= n - \lfloor n \rfloor \\ q(U_k) &= \Sigma\Delta_k (n - \lfloor n \rfloor) \end{aligned} \tag{2.19}$$

Rewriting equation 2.15 as

$$U_k = U_{k-1} + (x_{k-1} - q(U_{k-1})) \tag{2.20}$$

equations 2.18 and 2.20 together with 2.19 result in:

$$U_k = \frac{\Delta t_k}{T_{LO}} \tag{2.21}$$

This equivalence states that the delay between the reference and the divider output (that is the processed of the TDC) is directly proportional to the output of the $\Sigma\Delta$ accumulator. The amplitude of this delay sets a lower bound to the TDC full-scale requirements. Although the value of $\lfloor n \rfloor$ influences the U_k sequence, it is possible to derive general properties of U_k and therefore of Δt_k.

Starting from equation 2.15 the $\Sigma\Delta$ modulator dynamics, given a constant input x and an initial condition $U_0 = u_0$, can be summarized by the difference equations

$$U_k = \begin{cases} U_{k-1} + x - 1 & \text{if } U_{k-1} \geq \frac{1}{2} \\ U_{k-1} + x - 0 & \text{if } U_{k-1} < \frac{1}{2} \end{cases} \tag{2.22}$$

Assuming $\lfloor n \rfloor \neq 0$ (which means that n is fractional), you get from equation 2.19:

$$x = n - \lfloor n \rfloor \Rightarrow \begin{cases} x < 1 \\ x > 0 \end{cases} \tag{2.23}$$

Observe form eq. 2.22 that once U_k lies in the interval $[x - \frac{1}{2}, x + \frac{1}{2}]$, all future U_{k+j} will lie within the interval.

$$U_k \in \left[x - \frac{1}{2}, x + \frac{1}{2}\right] \quad \Rightarrow \quad U_{k+j} \in \left[x - \frac{1}{2}, x + \frac{1}{2}\right] \qquad \forall j \in \mathbb{Z}^+ \tag{2.24}$$

This sentence can be demonstrated by contradiction assuming U_k out of interval.

To begin observe from 2.22 and 2.23 that:

$$\begin{aligned} U_k < U_{k-1} \quad &\Rightarrow U_{k-1} \geq \tfrac{1}{2} \\ U_k \geq U_{k-1} \quad &\Rightarrow U_{k-1} < \tfrac{1}{2} \end{aligned}$$

Then assume that U_k exits the lower bound of the interval:

$$U_k < x - \frac{1}{2} \Rightarrow \begin{cases} U_k < U_{k-1} & \rightarrow U_{k-1} \geq \frac{1}{2} \\ U_{k-1} + x - 1 < x - \frac{1}{2} & \rightarrow U_{k-1} < \frac{1}{2} \end{cases} \tag{2.25}$$

resulting in a contradiction. Finally assume that U_k exits the upper bound of the interval:

$$U_k > x + \frac{1}{2} \Rightarrow \begin{cases} U_k > U_{k-1} & \rightarrow U_{k-1} < \frac{1}{2} \\ U_{k-1} + x > x + \frac{1}{2} & \rightarrow U_{k-1} > \frac{1}{2} \end{cases} \tag{2.26}$$

resulting again in a contradiction.

It has been demonstrated that $U_k \in [x - \frac{1}{2}, x + \frac{1}{2}]$. Recalling that $x \in [0, 1]$ we derive that $U_k \in [-\frac{1}{2}, +\frac{3}{2}]$. From eq.2.21 we finally get that:

$$\begin{cases} \Delta t_k & < +\frac{3}{2} T_{LO} \\ \Delta t_k & > -\frac{1}{2} T_{LO} \end{cases} \tag{2.27}$$

and therefore the minimum TDC full-scale when using a first order $\Sigma\Delta$ modulator is of $2 \cdot T_{LO}$.

2.3.2 Higher order $\Sigma\Delta$ modulators

Since the fractional error spectra energy is concentrated to higher frequencies ([12]), it is possible to take advantage of the system low-pass transfer function. Generating the N_k sequence with a high-order $\Sigma\Delta$ modulator, the spectral content of the TDC output signal is pushed to higher frequencies ([13]). The high-pass $\Sigma\Delta$ noise shaping, together with the low-pass transfer function form the TDC output to the LO control, allow to reduce the impact of fractional spurs.

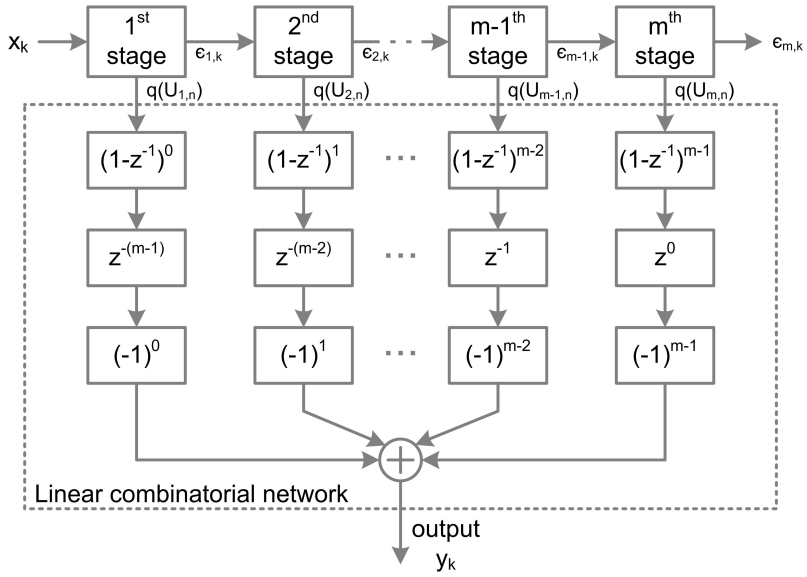

Figure 2.6: MASH $\Sigma\Delta$ modulator

High order $\Sigma\Delta$ modulator can be designed as a cascade of lower order (typically first order) modulators. This structure is known in literature as MASH (Multi-stAge noise-SHaping). Alternatively multi-loop $\Sigma\Delta$ modulator can be designed; but their stability is an issue for orders higher than two.

A block diagram of the MASH modulator is depicted in figure 2.6, while the single stage modulator scheme is depicted in figure 2.7. The signal to be

modulated x_k goes into the input of the first stage. The binary quantizer error $\epsilon_{m,k}$ of each stage is fed into the input of the following stage. The outputs of

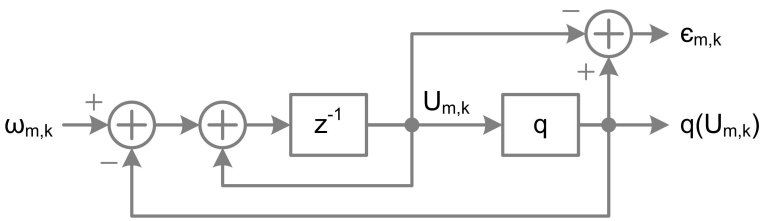

Figure 2.7: MASH single stage

all the stages are processed by a linear combinatorial network to produce the output of the MASH modulator. The linear combinatorial network is defined as:

$$y_k = \sum_{i=1}^{m} (-1)^{i-1} \cdot z^{i-m} \cdot \left(1 - z^{-1}\right)^{i-1} \cdot q(U_{i,k}) \tag{2.28}$$

Assuming $q(U_{m,k})$ to be in general uncorrelated with $q(U_{m+j,k-h})$ it is possible to evaluate the maximum peak to peak amplitude of y_k to be $2^m - 1$ for a m^{th} order MASH modulator.

To find what effects the MASH modulator will produce on Δt_k, consider that if the peak to peak amplitude of y_k is limited $2^m - 1$, the upper limit to the TDC full scale (eq. 2.13) results to be $(2^m - 1)\, T_{LO}$. This result sets only a theoretic upper bound and more accurate results can be obtained by simulating the designed modulator.

It is noticeable that this results underestimate the result obtained in previous section for the first order $\Sigma\Delta$ modulator (eq. 2.27). However this result highlight how much a high order MASH modulator can aggravate the TDC design specifications.

Another way to get to the same conclusion can be found in [13] where a theorem (*Theorem 1*) is demonstrated that states: "for an m-stage $\Sigma\Delta$ modulator with the linear combinatorial network defines as in 2.28, the output sequence can be represented as a sum of the m-step delay of the input x_k and an m^{th}

order difference of the binary quantizer error of the final (m^{th}) stage, as expressed in equation 2.29".

$$y_k = x_{k-m} + (-1)^{m-1} \cdot \left(1 - z^{-1}\right)^{m-1} \cdot \epsilon_{m,k} \tag{2.29}$$

The use of higher order $\Sigma\Delta$ modulators is very popular in analog PLLs, because it does not require any additional analog circuitry but a digital modulator. However it's impact on the TDC full-scale make it less attractive in the ADPLL system.

2.3.3 First order feedthrough $\Sigma\Delta$ modulator

It was demonstrated that 1st order $\Sigma\Delta$ modulator results in a minimum TDC full scale of $2T_{LO}$ (eq. 2.27). But, recalling the figure 2.4, it can be observed that after every reference rising edge, there is always one LO rising edge within one LO period. This sentence is self-proved by the definition of periodicity. This means that the first order $\Sigma\Delta$ modulator is not the best choice if the TDC full scale has to be minimized.

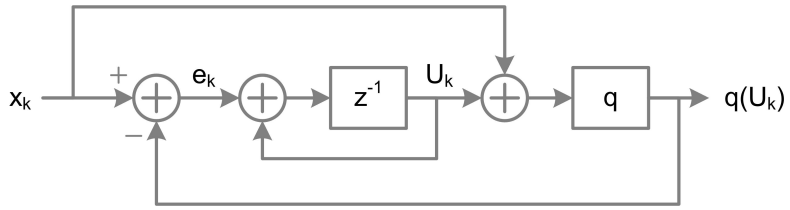

Figure 2.8: 1st order feedthrough $\Sigma\Delta$ modulator

In figure 2.8 the scheme of a 1st order feedthrough $\Sigma\Delta$ modulator is depicted. The difference equations of the 1st order $\Sigma\Delta$ modulator (eq. 2.22), can be rewritten as:

$$U_k = \begin{cases} U_{k-1} + x - 1 & \text{if } U_{k-1} + x \geq \frac{1}{2} \\ U_{k-1} + x - 0 & \text{if } U_{k-1} + x < \frac{1}{2} \end{cases} \tag{2.30}$$

Following the same procedure of subsection 2.3.1 it is possible to demonstrate that if U_k lies in the interval $[-\frac{1}{2}, \frac{1}{2}]$, all future U_k will lie within the same

interval. Equation 2.24 can be rewritten as:

$$U_k \in \left[-\frac{1}{2}, +\frac{1}{2}\right] \quad \Rightarrow \quad U_{k+j} \in \left[-\frac{1}{2}, +\frac{1}{2}\right] \qquad \forall j \in \mathbb{Z}^+ \tag{2.31}$$

Therefore the use of a 1st order feedthrough $\Sigma\Delta$ modulator results in a minimum TDC full-scale of only one T_{LO}.

To highlight the difference between the feedthrough and the standard $\Sigma\Delta$ modulator, a simulation has been performed with a full scale sinusoidal input x_k. The simulation results are depicted in figure 2.9. It can be seen that even if the $q(U_k)$ in the two cases are very similar, the U_k of the standard $\Sigma\Delta$ modulator has a double peak to peak amplitude with respect to the feedthrough topology.

Furthermore in the case of the standard $\Sigma\Delta$ modulator it can be seen that U_k contains x_k accordingly to equation 2.24.

2.3.4 Fractional ADPLL with Digital compensation

It was shown that the use of high order $\Sigma\Delta$ modulator to suppress fractional spurs affects the TDC design constraints. An alternative approach is to compensate the fractional errors. This concept has been developed in the seventies [3] and it was developed for analog PLLs ([14]). The target is to cancel PD output deterministic components, which are caused by the division sequence N_k, and thereby avoid phase jitter in the VCO.

For a given N_k sequence it is possible to evaluate the compensation sequence Δt_c with equation 2.13. The compensation sequence Δt_c may be conveniently obtained directly from the accumulator output of a first order $\Sigma\Delta$ modulator.

In an analog PLL a DAC[5] is required to generate the compensation sequence to be added to the PD output. The complexity of this solution within an analog PLL has caused its little success. However the presence of a digitalized phase error information (the TDC output) in the ADPLL allows to apply this concept

[5]Digital to Analog Converter

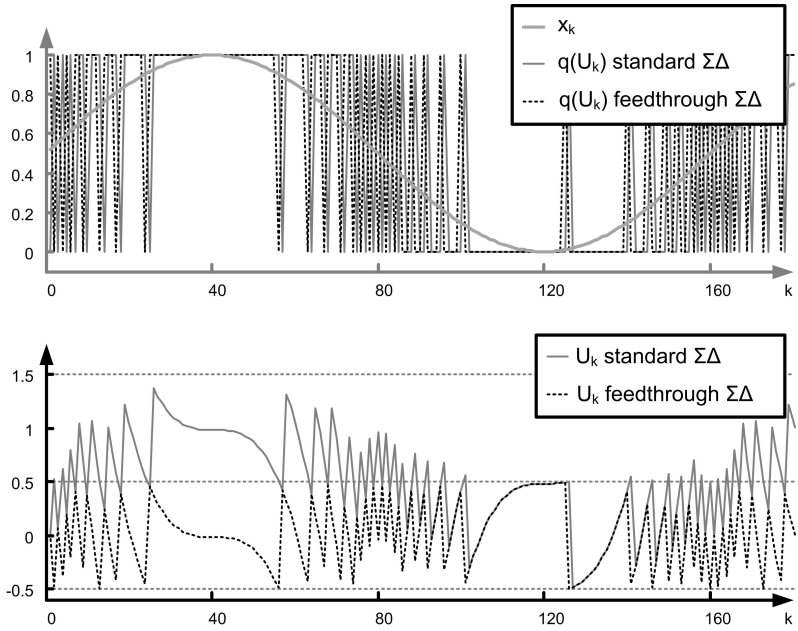

Figure 2.9: Feedthrough and standard $\Sigma\Delta$ modulators simulation

almost for free. A schematic representation of the fractional errors compensation is depicted in Fig. 2.10. No additional analog circuitry is required but only a digital subtracter.

As in the Analog Case, the effectiveness of the fractional errors compensation is based on the matching between two different signal paths: the first that from the modulator pass trough the divider and the TDC, the second that goes directly from the modulator to the subtracter. The TDC gain k_{TDC} is prone to errors due to its analog nature, and therefore can unbalance the two paths. To overcome this limitation, a TDC gain calibration algorithm has been used (it will be discussed in chapter 3). Even if calibration is performed, the TDC gain will always be affected by a residual error. The capability of the compensation to mitigate the fractional spurs depends on the accuracy of the gain calibration.

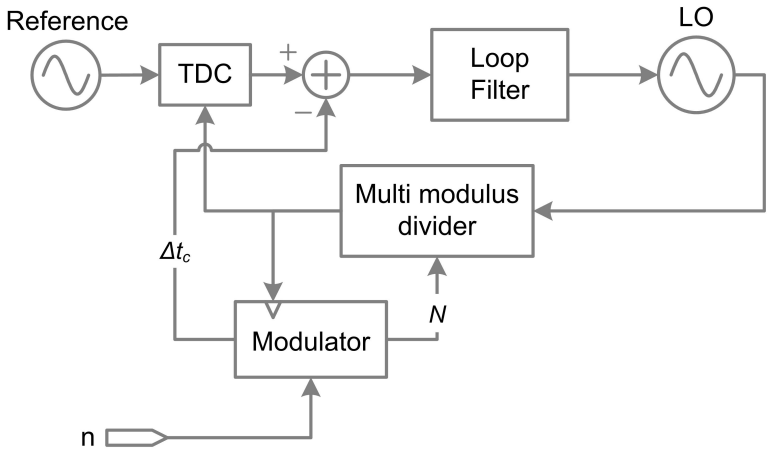

Figure 2.10: Digital compensation scheme

The concepts of compensation and $\Sigma\Delta$ modulation may also be used together, benefiting the LO spectral purity. However as demonstrated in section 2.3.2 the use of high order $\Sigma\Delta$ modulator to drive the multi-modulus divider has a significative drawback: it requires an extended full scale for the TDC.

The signal that has to be processed by the TDC may be divided in two different components: the LO phase fluctuations and the fractional errors. The phase fluctuations depend on the spectral purity of the LO; all the wireless standards require the Root Mean Square (RMS) value of the LO phase fluctuation to be a small fraction of radians. On the contrary the fractional errors magnitude depends on the multi-modulus divider resolution and on the divider control sequence (as shown in previous sections), and its RMS value is generally of several LO radians. The phase fluctuations represent the TDC signal component of interest that has to be controlled by the system, while the fractional errors are just deterministic disturbances. Even if the fractional errors are not of interest, the TDC has to be designed to detect them without incur in saturation (or other nonlinearities); otherwise also the components of interest would be deteriorated. As a result, the fractional errors limit the min-

imum TDC full scale. The TDC resolution has to be designed to meet the LO spectral purity requirements as shown in section 2.1.1. These points allow the definition of the TDC design requirements starting from the system specifications: the design of the $\Sigma\Delta$ modulator sets a lower bound to the TDC full scale, while the LO spectral purity requirements sets the TDC resolution.

The TDC dynamic range (the ratio between full scale and resolution) has a direct impact on the TDC implementation costs (area, power consumption) independently on the adopted architecture. The target of a digital intensive approach to the frequency synthesis has been pursued choosing a first order $\Sigma\Delta$ modulator relaxing the design specification.

2.4 Direct modulation

A frequency synthesizer is commonly required for carrier generation in telecommunication systems. Furthermore it is possible to directly synthesize a phase-frequency modulated signal. In order to modulate the LO output phase the modulation signal can be added to the loop filter input, the signal is therefore filtered by low pass transfer function $H(s)$.

Wireless standards continue to propose growing data rates and consequently wider band signals. In chapter 1 it has been shown that the PLL bandwidth directly affects the LO spectral purity. Literature proposes more complicated modulation schemes that allow to overcome the trade off between spectral purity and modulation bandwidth.

2.4.1 Two-point modulation

The two-point modulation is based on the injection of the modulation signal in two different nodes of the system (Fig. 2.11). The transfer functions H_a and H_b obtained injecting a signal in a and b, and observing ϕ_o are:

$$H_a(s) = \frac{\phi_o(s)}{a(s)} = \frac{F(s) \cdot LO(s)}{1 + G(s)} \tag{2.32}$$

$$H_b(s) = \frac{\phi_o(s)}{b(s)} = \frac{LO(s)}{1 + G(s)} \tag{2.33}$$

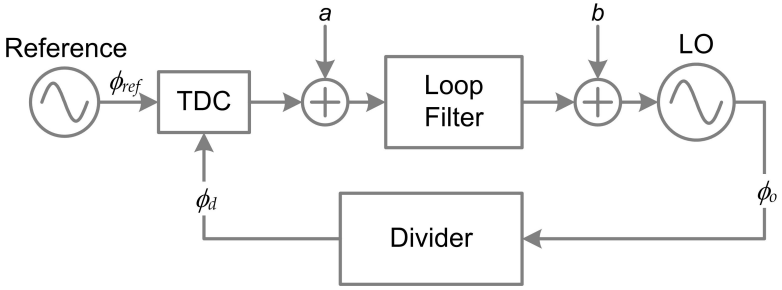

Figure 2.11: Two-point injection nodes

$$G(s) = \frac{k_{TDC} \cdot F(s) \cdot LO(s)}{N} \tag{2.34}$$

Where $F(s)$ and $LO(s)$ are respectively the Loop Filter and the Local Oscillator transfer functions, while k_{TDC} and N are the TDC gain and the division factor. Assume to generate a and b signals from a modulation signal ϕ_m, such that:

$$a(s) = \phi_m(s) \cdot \frac{k_{TDC}}{N} \tag{2.35}$$

$$b(s) = \phi_m(s) \cdot \frac{s}{k_{LO}} = \phi_m(s) \cdot \frac{1}{LO(s)} \tag{2.36}$$

It is therefore possible to evaluate the transfer function between ϕ_m and ϕ_o applying the superposition principle.

$$\begin{aligned} \frac{\phi_o}{\phi_m} &= a(s) \cdot H_a(s) + b(s) \cdot H_b(s) \\ &= \frac{k_{TDC}}{N} \frac{F(s) \cdot LO(s)}{1 + G(s)} + \frac{1}{LO(s)} \frac{LO(s)}{1 + G(s)} \\ &= \frac{G(s)}{1 + G(s)} + \frac{1}{1 + G(s)} \\ &= 1 \end{aligned} \tag{2.37}$$

The identity obtained with equation 2.37 shows that the modulation signal

is not filtered anymore and therefore this solution overcomes the bandwidth limits.

The a and b signals must be generated following equations 2.35 and 2.36. It is noticeable that equation 2.35 is just to a proportional coefficient, while equation 2.36 has a derivative coefficient that relate the ϕ_m information to the control frequency of the LO. This difference is due to the different points of injection: while a is added to the phase error node, b is added to the LO frequency control word; therefore the derivative relation between frequency and phase has to be respected.

Two analog parameters occur in those equations: k_{TDC} and k_{LO}. Any error in k_{TDC} and k_{LO} will affect the accuracy of the two-point modulation, resulting in undesirable fluctuations of the modulation transfer function. Therefore those parameters have to be calibrated; this topic will be addressed in chapters 3 and 4.

Impact of two-point modulation on the TDC design

The impact of the two-point modulation on the TDC design can be argued evaluating it effects on ϕ_e, which is defined as the difference $\phi_{ref} - \phi_d$ (equation 1.2). Recalling the equations 2.32, 2.33 and 2.34, the transfer functions E_a and E_b obtained injecting a signal in a and b, and observing ϕ_e can be expressed as:

$$E_a(s) = \frac{\phi_e(s)}{a(s)} = \frac{F(s) \cdot LO(s)}{N\,(1 + G(s))} = \frac{H_a(s)}{N} \tag{2.38}$$

$$E_b(s) = \frac{\phi_e(s)}{b(s)} = \frac{LO(s)}{N\,(1 + G(s))} = \frac{H_b(s)}{N} \tag{2.39}$$

It is therefore possible to evaluate the overall transfer function between ϕ_m and ϕ_e applying the superposition principle.

$$\begin{aligned} \frac{\phi_e}{\phi_m} &= a(s) \cdot E_a(s) + b(s) \cdot E_b(s) \\ &= \frac{1}{N}\left[a(s) \cdot H_a(s) + b(s) \cdot H_b(s)\right] \\ &= \frac{1}{N} \end{aligned} \tag{2.40}$$

The equation 2.40 states that the two-point modulation signal has a flat transfer function to the phase error. This means that the phase modulation produce a new contributor to the TDC input signal, which is added to the LO phase fluctuations and the fractional errors. Therefore the TDC input dynamic range has to be extended to avoid saturation effects produced by the two-point modulation contribution.

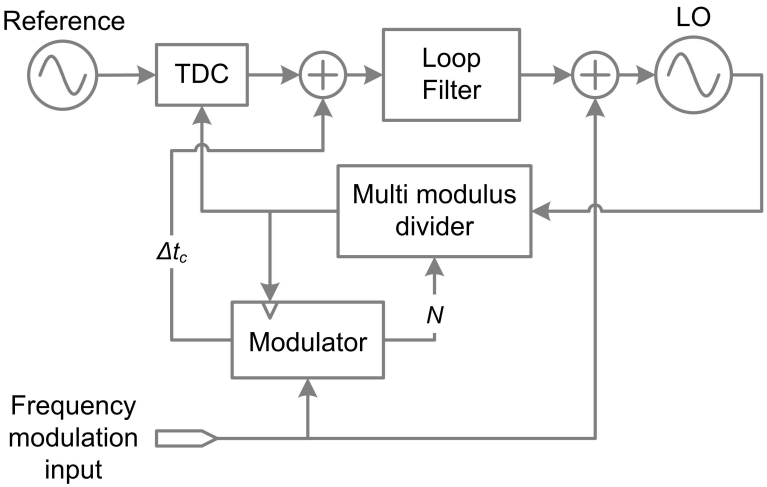

Figure 2.12: Two-point modulation and Fractional-N compensation

A convenient way to overcome this additional requirement to the TDC dynamic range is to apply the two-point modulation as depicted in figure 2.12. The modulation signal in this scheme represents a frequency modulation, therefore it can be directly added to the LO frequency control word (gain blocks required for adjust proportional coefficients are not represented for simplicity). The frequency modulation word is processed by the modulator that produce N and Δt_c sequences. The frequency modulation signal can be generated from any phase modulation signal with a derivative transfer function:

$$f_m(s) = \frac{2\pi}{s}\phi(s) \tag{2.41}$$

Since the frequency modulation signal f_m is applied to the modulator, the divider control sequence N takes into account for the modulation signal effects. In section 2.3.3 it was demonstrated that whatever input is applied to the 1st order feedthrough $\Sigma\Delta$ modulator, the resulting ϕ_e spans a range of only one T_{LO}. In this way the TDC full scale has not to be extended to perform the modulation.

Also the fractional error compensation sequence Δt_c is affected by the frequency modulation signal. It was shown that the Δt_c sequence is related to the integral of the modulator input signal (section 2.3.1). Therefore the Δt_c sequence contains the phase modulation information. Since the Δt_c sequence is injected to the system in the same point of a (figures 2.11 and 2.12), the phase modulation information is correctly added to the system. Therefore the solution of figure 2.12 allows to apply the two-point modulation scheme overcoming its impact on the TDC design.

Chapter 3

The Time to Digital Converter

Contents

The Time to Digital Converter (TDC) is a particular type of Analog to Digital Converter (ADC) where the analog input is the timing difference between two events. In the evolution of phase-locked-loops towards a more flexible digital architecture, the TDC represents one of the possible bottlenecks in the design of a spurs-free and low noise solution. The TDCs are also required in a number of different applications (Nuclear physics experiments, Light Detection And Ranging systems and other more), each application sets different requirements and constraints on the TDC design.

Different architectures of TDC were proposed in literature and some of them are discussed in the first part of this chapter. The two-dimensions Vernier TDC (a particular architecture of TDC [19]) will be deeply investigated in the

second part of the chapter, together with design details about the realized TDC prototype.

3.1 The digitalization of time intervals

The analog to digital conversion can be divided into three basic operations: sampling, quantizing and coding. In standard ADCs the sampling operation is performed by a sample and hold circuit, which stores the sample for a limited time (usually one ore more reference periods). The availability of the sample for such a time allow to perform different operations on it, this is the key factor that has allowed the great variety of ADC architectures.

In TDCs the analog input signal is the delay between two events. This delay signal has different properties from conventional analog signals. First of all it is a sampled signal, since it is defined as the delay between singular events. Furthermore it is not possible to store it in a simple and reliable way. The amplification, that is an operation widely used in ADCs, is nontrivial to be implemented on the delay between two events. This limitations have produced a small variety of TDC architectures.

Most of the TDC architectures are similar to the flash ADC topology, where a number of references is generated (for example with a string resistor ladder in a voltage flash ADC) that are compared to the input signal. In this case the number of references and comparators is the same as the number of quantization levels.

Few TDC architectures have been demonstrated in ADPLL applications; they are briefly discussed in the next sections.

3.1.1 The single delay line TDC

Probably the most intuitive example of the flash TDC approach is the single delay line TDC. A block and a timing diagram of a single delay line TDC are reported in figure 3.1. In this architecture the reference scale is generated through a delay line, and the time references are the delays introduced by the taps of a delay line.

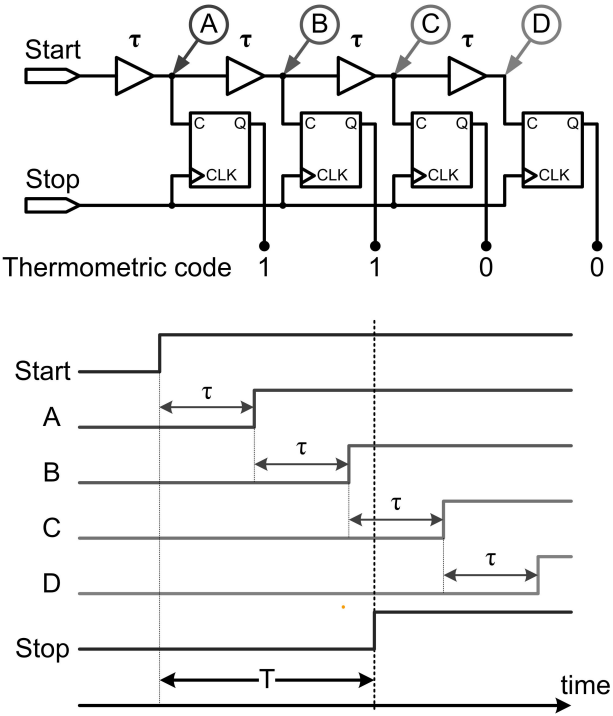

Figure 3.1: Single delay line TDC

The delay line produces a number of reference events that are uniformly spaced in time from the **start** event. Time comparators are used to detect if the **stop** event happens before or after than each reference event. Each time comparator can be thought as a single bit TDC and can be designed as a flip-flop.

The TDC output is the vector of time comparators outputs, which rappresent a thermometric code (as it is for flash ADCs). To obtain a binary code a thermometric to binary conversion has to be performed ([1]).

The resolution of this type of TDC is equal to the single stage delay τ. This is the major drawback of this solution since it results in a technology limitation

of the TDC resolution. If the targeted resolution is below the minimum reliable delay achievable by the technology, this architecture cannot be used.

Current CMOS technology offers very short inverter delay values, and single delay line TDC have been demonstrated with resolution down to $20ps$ ([16] proposes a TDC design targeted to an ADPLL).

3.1.2 The Vernier TDC

Vernier TDC allows to overcome the technology limitation on time resolution. In a Vernier TDC two delay lines are used to delay both input signals, as depicted in figure 3.2.

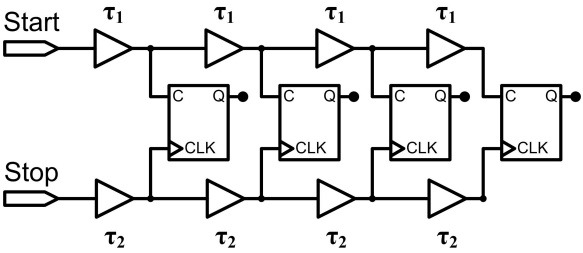

Figure 3.2: Vernier TDC

In a Vernier scale (figure 3.3) a differential delay Δ between two corresponding taps of the lines is accumulated after each stage so that a signal edge, which lags a reference edge by $n \cdot \Delta$ at the input of the Vernier, will be lined up with it after n stages.

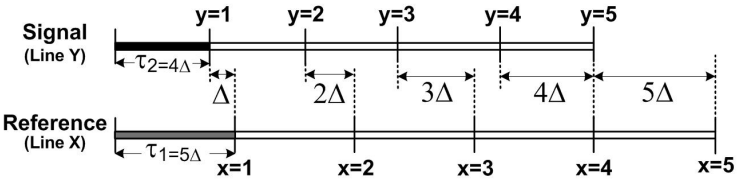

Figure 3.3: Vernier scale

Since the reference scale is made of relative delays, the TDC time resolution is not technology limited by the smallest reliable delay available in the technology. The TDC time resolution is limited only by the accuracy of the delay element, which can be arbitrarily enhanced at cost of area and power.

It has to be noticed that this architecture still belongs to the flash ADC class. Even if two delay lines are used, there is still only one reference scale (made of relative delays); and each reference is compared to the "sample" by a dedicated time comparator.

Some variations were proposed to the Vernier principle. Tonietto ([17]) has proposed a TDC solution that replaces the delay line connected to the signal input, with the periodicity of the input signal (the radio-frequeny LO output signal). This solution was also recently applied to an ADPLL presented in [18].

3.1.3 The Gated Ring Oscillator TDC

A schematic rapresentation of a Gated Ring Oscillator TDC (GRO TDC) is depicted in figure 3.4.

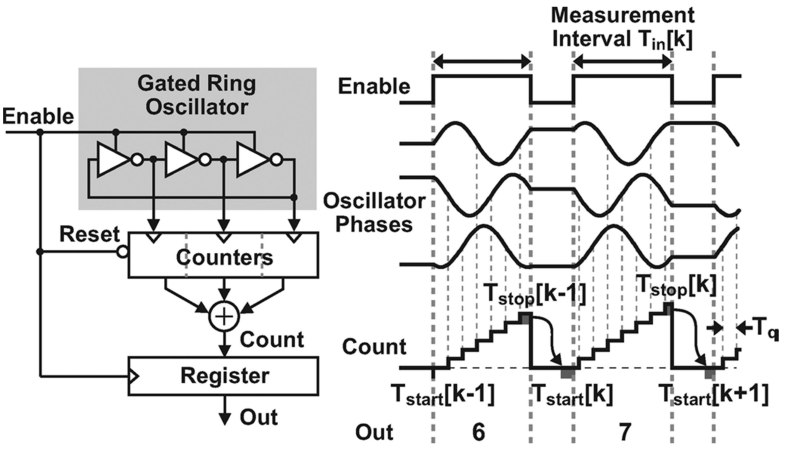

Figure 3.4: Gated Ring Oscillator TDC (GRO TDC)

The main claim of this solution [21] is to produce a quantization noise shaping (a concept usually related to $\Sigma\Delta$ ADC architectures).

The time references that are used are generated through a ring oscillator that is enabled only during the time interval that has to be measured. After the measurement the oscillator is disabled freezing all its state variables.

The benefit of gating the oscillator is that the quantization error occurring at the end of a given measurement interval is transferred to the next measurement interval. It can be shown ([21]) that this results in a first order quantization noise shaping.

Although the GRO quantization noise shaping suggests some relationship with $\Sigma\Delta$ ADCs, the GRO is still a flash converter. The particularity of the GRO is that its reference scale is not constant, but is time varying. The quantization noise shaping is originated by the time varying reference scale.

3.2 Two-Dimensions Vernier TDC

In a Vernier TDC (section 3.1.2) the delay quantization is realized by taking the time differences between taps located in the same position of the two delay lines (e.g. the delay difference between the 2^{nd} stage of first line and the 2^{nd} stage of second line).

If all possible relative delays among the taps (figure 3.3) are considered, the resulting quantization levels can be conveniently represented on the Vernier plane ([20]) of figure 3.5, where 25 quantization levels instead of 5 are produced. However only a part of the generated quantization levels is uniformly spaced.

Nevertheless an extension of the TDC range from $[\Delta, 5\Delta]$ to $[-3\Delta, 9\Delta]$ is obtained (grey zone). This multiplies almost by three the number of uniformly spaced quantization levels of the linear Vernier.

The Vernier plane can be considered as an extension of both the single line TDC and the Vernier TDC, which lie respectively on the borders and on the diagonal of the plane in figure 3.5.

If a greater part of the Vernier plane is taken into account (figure 3.6), a wider range of uniformly spaced quantization levels is generated. The number

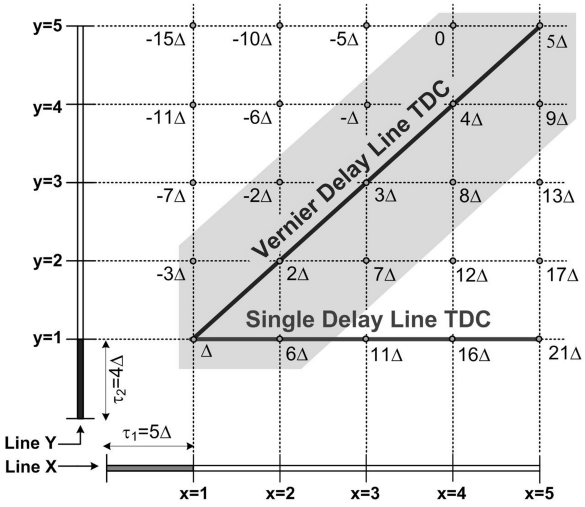

Figure 3.5: The Vernier Plane

of uniformly spaced quantization levels is grater than half of the total number of points of the plane.

The quantization levels that appear on the Vernier plane depends on the ratio between the delays of the lines X and Y. Different values of τ_1/τ_2 produce different Vernier Planes. To guarantee the generation of uniformly spaced quantization levels, the τ_1/τ_2 ratio has to be proper designed.

A two-dimensions Vernier TDC implementation can be derived starting from the classical linear Vernier architecture where the time references are realized with two delay lines and the digital conversion is performed by flip-flops used as time comparators. In figure 3.7 a circuit implementation of the two-dimensions Vernier is shown. The scheme refers to an integrated prototype, measurement results are discussed in chapter 6. In this case, the ratio between the tap delay of the two lines τ_1/τ_2 is equal to $11/10$. Using 19 stages for line X and 11 stages for line Y, 119 uniformly spaced quantization levels are obtained.

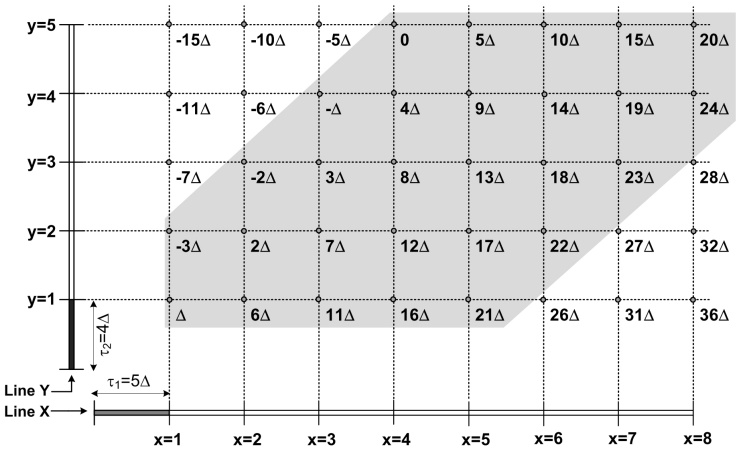

Figure 3.6: Extension of the Vernier Plane

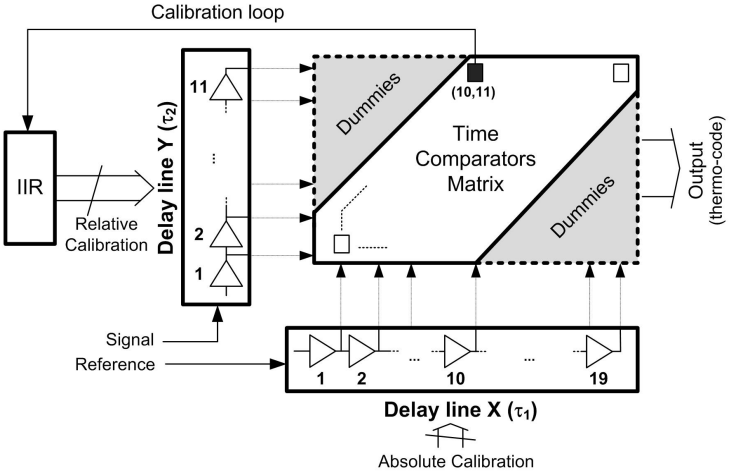

Figure 3.7: Two-dimensions Vernier TDC

3.2.1 Two-Dimensions TDC design

To validate the solution proposed, a TDC prototype was integrated in 65nm CMOS technology, with a target time resolution of 5ps and a full scale of 595ps (119 quantization levels). The technology used gives a minimum delay per stage of 20ps if worst case temperature and process variations are considered. As a consequence, a Vernier architecture (either traditional or two-dimensions) became necessary to reach the target resolution. The design of the prototype resulted in nominal values for the two delay lines of 50ps and 55ps. The entire TDC full scale is covered using 19 element for line X, 11 elements for line Y resulting in a total delay for line X of 1.045ns, that is less than twice the TDC full scale.

The TDC design starts from the time comparators whose analog performances (jitter, offset, matching) are almost independent of the other TDC building blocks. The input capacitance of the time comparators has to be minimized since in a two-dimensions Vernier an entire column (or row) of comparators is connected to a single stage of the delay lines. This sets the required driving capability of the delay element and at the same time affects the total power consumption of the two delay lines. Once time comparators are designed, the delay lines can be realized, with the target of minimizing the jitter noise and the distortion introduced. In the following, each TDC building block will be discussed, focusing on the main parameters that influence the final performance.

The time comparator

The time (or phase) comparator can be thought as a single bit TDC which decides if the delay between the input signals is positive or negative (identifying which signal corresponds with the first positive edge).

The time comparator is functionally equivalent to a voltage comparator in a flash ADC. As in a voltage comparator the input signal is a voltage difference, in a time comparator the input signal is a delay between two rising edges (i.e. a time difference).

The most important parameter for the design of this block is the offset matching. While for a voltage comparator the offset is represented as a voltage

source in series to one of the comparator inputs, in this case the offset corresponds to an additional delay between the input signals and it is known as time skew. The offset matching among the comparators affects the TDC linearity with a direct impact on the DNL (its influence on INL is less significant since the error introduced is not accumulated during the quantization process).

The time comparator was implemented with a Set-Reset (SR) latch (figure 3.8). The solution was preferred over the flip-flop-D due to its perfect symmetry that allows to minimize systematic time skew and to provide a symmetric load to both delay lines.

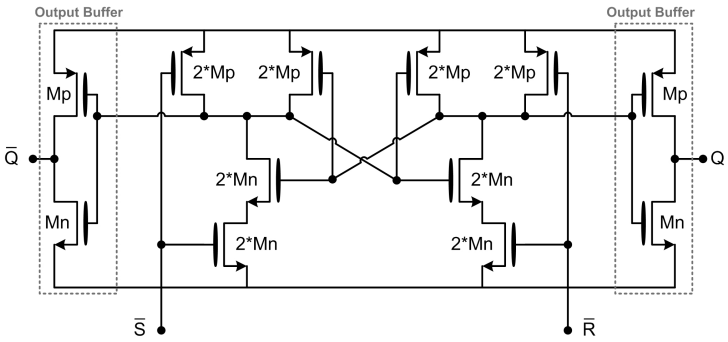

Figure 3.8: Set-Reset latch

Furthermore, the SR latch results in a very compact layout, highly desirable in a two-dimensions TDC where a large number of comparators are arranged in a bi-dimensional structure. The size of the latch input devices was designed exploiting Montecarlo simulations to optimize offset mismatches. A histogram of the simulated offset is reported in figure 3.9. The resulting time comparator exhibits a standard deviation of the input referred offset of less than 1ps (enough for the target resolution of 5ps).

Digitally controlled delay elements

Two different delay elements are needed to create the "time reference" for the Vernier plane and their relative accuracy determines the linearity of the

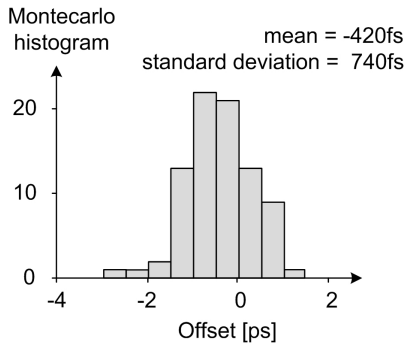

Figure 3.9: Set-Reset latch offset matching

TDC. The delay element circuit is shown in figure 3.10.

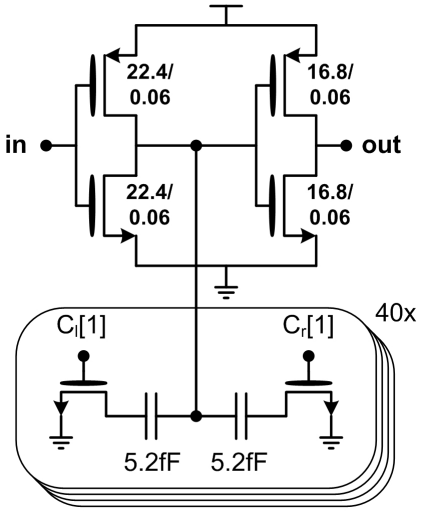

Figure 3.10: Digitally controlled delay element

It is made up by the cascade of two inverters plus a digitally controlled

capacitor bank used for the delay calibration. Two cascaded inverters in each delay tap are required when using an SR latch as a time comparator since such a latch operates only on the rising edges of the input signals.

The tuning of the delay is performed acting on the capacitive load (at the output of the first inverter), this solution results in a linear tuning characteristic (in term of digital word vs. delay). Finally, the presence of two delays stage per tap (required to have a non-inverting delay element) offers isolation between the capacitor bank and the latch, allowing to realize sharper edges in front of the latch.

To reach the target resolution of 5ps, the delays in line X and Y have to be set respectively to 55ps and 50ps. To compensate for Process, Voltage and Temperature (PVT) variations a capacitor bank composed of 80 elementary Metal-Oxide-Metal capacitors (MOM) is used. They can be either connected to ground through an n-MOS switch or left floating, producing in the worst-case PVT corner a delay tuning with steps of 1ps. The simulations of the tuning characteristics for the most significant PVT variation corners are reported in figure 3.11. From the plot it is possible to verify that the required delays of 50ps and 55ps can be obtained in all the corners.

3.2.2 Relative calibration

For the Vernier plane to be proper defined, the ratio τ_1/τ_2 needs to be precisely set to its designed value. The design of a τ_1/τ_2 ratio equal to $k/(k-1)$ allows to make the most of Vernier plane properties ([20]). This ratio can be controlled using a delay-locked loop (DLL) which forces the total delay accumulated after k-1 stages of line X to be equal to the one accumulated after k stages of line Y.

Both delay lines are fed with the same signal during the calibration phase, and the output of the latch at the position $(10, 11)$ is used to estimate the delay difference between 10 delays of line X and 11 delays of line Y. The latch output signal is integrated by an infinite impulse response (IIR) filter whose digital output is directly used to control the delay elements of line Y (this is possible since the delay lines are digitally controlled), forcing the τ_1/τ_2 ratio to be $10/11$. Since this calibration requires the same signal in both delay

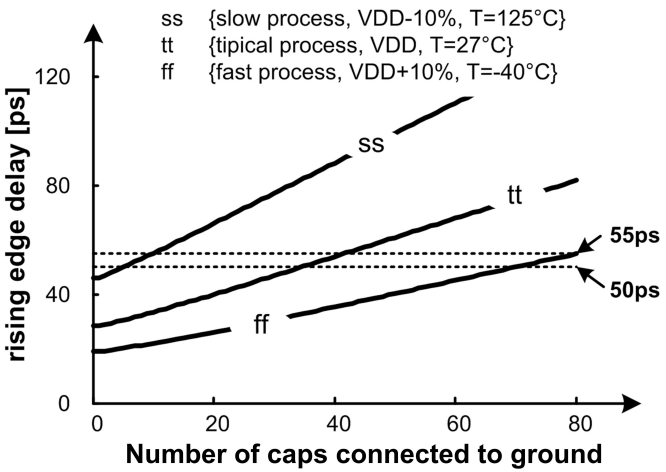

Figure 3.11: Delay element tuning characteristics

lines, this solution demands an input network that is able to interleave between acquisition and calibration phases.

This calibration algorithm has been named "relative" to distinguish it from the gain calibration. The gain calibration (section 3.3) sets the absolute values for the gain and the time resolution of the TDC (and is therefore also named "absolute" calibration), while the relative delay sets the ratio τ_1/τ_2 and is required to improve the TDC linearity.

Input network

The input network designed for the TDC prototype ([20]) is depicted in fig.3.12. The generation of the two operative phases is performed injecting a double frequency clock that is divided by two producing quadrature outputs. Although this solution guarantees a robust generation of the two operative phases it is not applicable in an ADPLL. The TDC calibration can not justify the request of a double frequency clock.

Therefore the input network has been redesigned for the ADPLL prototype

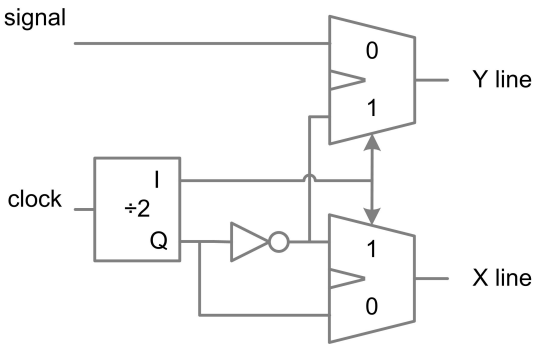

Figure 3.12: First prototype's input network

allowing the generation of calibration and acquisition phases without a double frequency clock (figure 3.13). The idea in this case is to use both the clock rising and falling edges, respectively for the acquisition and calibration. The Delay line X is used to determine the end of the acquisition process. Once the reference rising edge appears at the end of the line, the multiplexers are switched. The reference signal is then inverted to generate the calibration edge.

An enable signal, that has to be active for the calibration phase to be generated, has been added. It is then possible to perform the calibration at a lower rate than the reference clock, resulting in a reduced power dissipation related to the calibration. Since the calibration phase is generated when a positive transition appears at the end of the delay line X, the enable signal can be produced by sequential logic clocked by the reference. As it can be seen in figure 3.14, an additional advantage of this solution is that since the tune signal changes only during the calibration phase, it cannot perturb the acquisition operations.

Delay lines control

The target of the relative calibration is to lock the relative delay between the two lines with a time error that is smaller than the TDC resolution. It is also

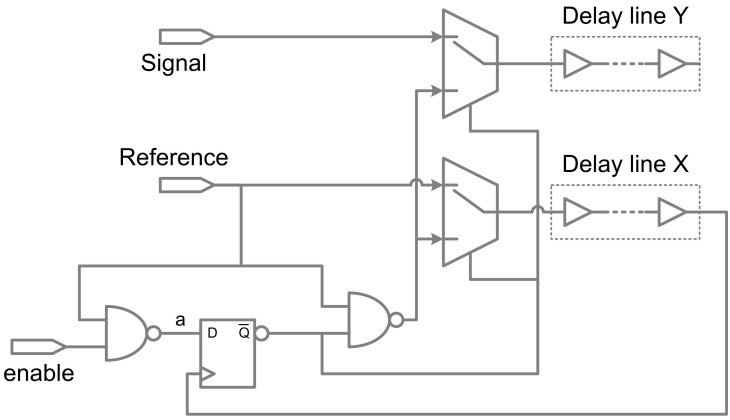

Figure 3.13: Redesigned input network

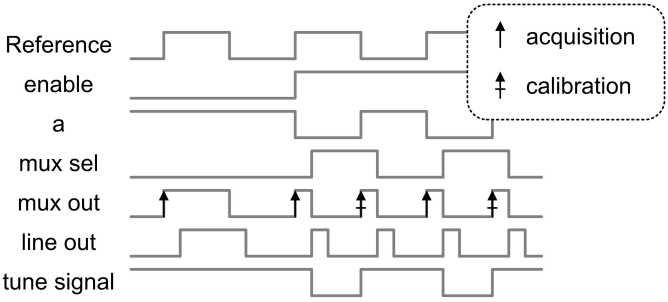

Figure 3.14: Redesigned input network time diagram

desirable that the nominal value of the single delay element is compensated against PVT variations. In a standard CMOS technology, PVT variations have a considerable impact on an inverter delay. In the $55nm$ technology adopted for the ADPLL prototype, the delay may vary of $+75\% - 30\%$ according to simulations. These information, applied to a delay line with a nominal total delay of $550ps$ that has to be controlled with pico-seconds resolution, results in a big dynamic range.

If all the digitally controlled delay elements are controlled by the same word, the single delay resolution is equal to the total resolution divided by the number of stages. Moreover the single delay element dynamic range is the same of the whole line (as both the resolution and the full scale are divided by the number of stages).

In order to relax the single delay element dynamic range, a "modulated" approach has been adopted: the delay elements of the delay line are not equally controlled. This techniques is depicted in figure 3.15: if the total delay has to be increased of one LSB, only one delay element is increased of one LSB. The dynamic range of the single delay element is relaxed by a factor equal to the number of delay elements in the line.

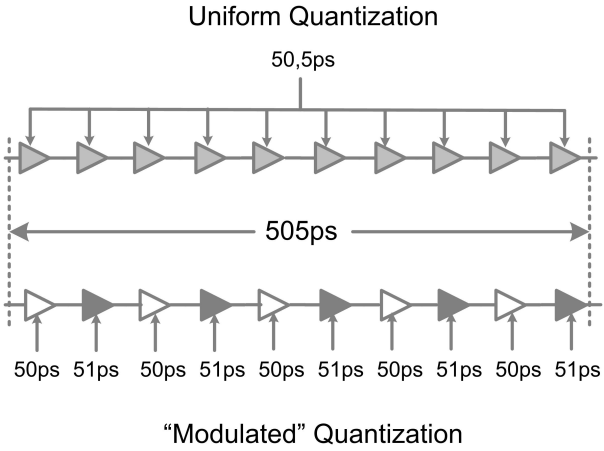

Figure 3.15: Delay line elements modulated control

The term "modulated" is used to describe this solution because of its likeness to the $\Sigma\Delta$ modulation principle: both techniques allow to obtain a high resolution from the cumulative effect of coarse samples. The proposed techniques cumulates the effect of samples generated by different elements, while $\Sigma\Delta$ modulation cumulates the samples in the time domain.

This solution has a significative drawback: it introduces a deterministic non-ideality in the system. This non-ideality has a direct effect on the TDC linear-

ity, its effects can be analyzed trough simulations. However a conscious design of the TDC allows to keep this effects under control.

3.2.3 Noise and linearity analysis

The non-idealities of the two delay lines limit the effective number of bits achievable by the TDC. To quantify this effect the delay must be modeled in a way that accounts for noise and mismatches. The model used for the analysis of the two-dimensions Vernier TDC is reported in figure 3.16. The "analog

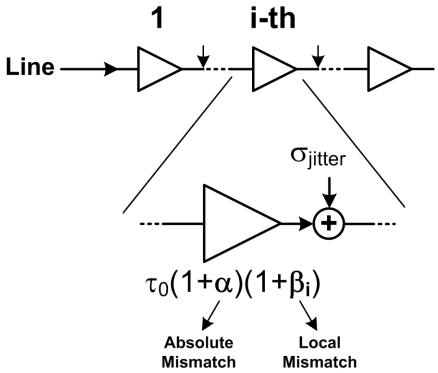

Figure 3.16: Delay model including non-idealities

jitter noise" introduced by each stage is represented inserting after each delay stage an additive, Gaussian and uncorrelated noise source with a standard deviation equal to σ_{jitter}. In addition, temperature and process variation are modeled with parameters α and β_j. The parameter α is associated with the absolute delay variation that affects all the elements in the same way, while β_j corresponds to the local mismatch of each element that changes along the lines. The model shown in figure 3.16 is described by the following equation:

$$\tau_j = \tau_0 \left(1 + \alpha\right) \left(1 + \beta_j\right) + \sigma_{jitter} \tag{3.1}$$

Jitter noise analysis

The jitter noise introduced by each stage limits the minimum delay that can be reliably measured by the TDC since adds an uncertainty to the measure. In particular, to guarantee less than one LSB integral error, the maximum jitter accumulated along the delay lines has to be lower than the TDC resolution. The variance of the jitter noise $\sigma_{jitter}(x, y)$ accumulated at a generic position (x,y) in the Vernier plane can be evaluated using the model described by equation 3.1 (with $\alpha = 0$ and $\beta_j = 0$) obtaining

$$\sigma^2_{jitter}(x, y) = \sum_{i=1}^{x} \sigma^2_{x,i} + \sum_{i=1}^{y} \sigma^2_{y,i} = x \cdot \sigma^2_x + y \cdot \sigma^2_y \tag{3.2}$$

where σ^2_x and σ^2_y are the variances of the jitter noise introduced by each stage of lines X and Y respectively. Assuming a total number of quantization levels equal to N, in the solution proposed the total numbers of elements for the two delay lines X, Y are respectively the superior integer of $2\sqrt{N/2}$ and $\sqrt{N/2}$ which lead to a maximum accumulated jitter of

$$max\left(\sigma^2_j itter\right) = 3\sqrt{\frac{N}{2}}\left(\sigma^2_x + \sigma^2_y\right) \tag{3.3}$$

Compared to the standard Vernier TDC (where the number of element of both delay lines is equal to N), the proposed solution reduces the accumulated jitter noise by a factor proportional to $\sqrt{N}$ assuming the same current is used in the delay elements in the two cases. For a given power consumption, shorter delay lines allow using much more current per stage that gives smaller jitter noise sources further reducing the accumulated noise.

Linearity analysis

The linearity of standard and two-dimensions Vernier TDCs were characterized in terms of Integral Non Linearity (INL) and Differential Non Linearity (DNL) including the effect of the relative calibration loop. This was realized performing Montecarlo simulations of a behavioral model described by equation 3.1, assuming a variance σ^2_α for α and σ^2_β for β_j, while setting $\sigma^2_{jitter} = 0$.

The comparison was done for the same number of quantization levels of the prototype ($N = 119$). The INL variance σ^2_{INL} as a function of the TDC output code i is plotted figure 3.17.a (normalized to the mismatch variance σ^2_β). Compared to the standard approach (dashed line), the two-dimensions

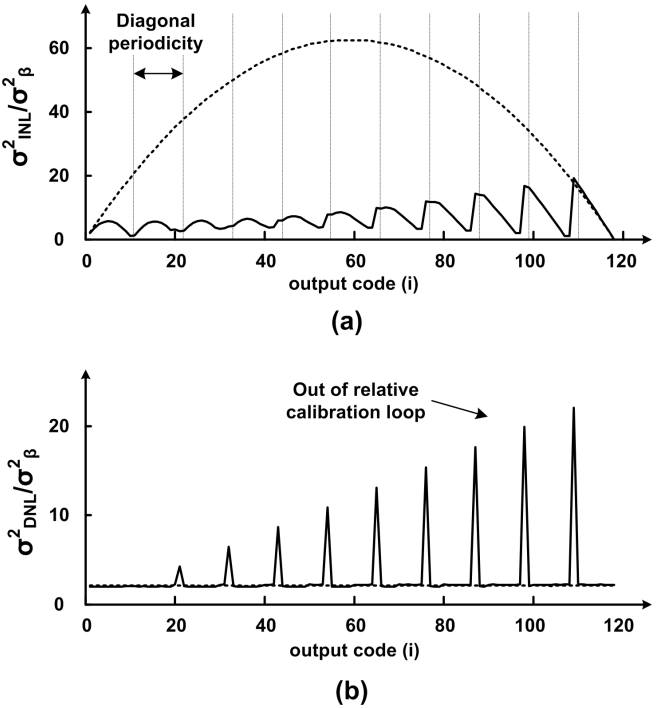

Figure 3.17: Theoretical linearity comparison of standard (dashed line) and two-dimensions Vernier (solid line) TDCs

Vernier solution (solid line) reduces by approximately a factor of three the maximum INL thanks to the use of shorter delay lines. In fact, while the standard Vernier TDC needs two lines of 119 elements each, the two-dimensions Vernier requires a line X with 19 elements and a line Y with 11 elements.

The exploration of a plane, instead of a line, gives an equivalent folding of

the time quantization realized along the two delay lines. This produces a periodicity in the INL that depends on the number of consecutive quantization levels that lie on each diagonal (in this case equal to 11). The relative calibration algorithm reduces the impact of delay elements mismatches. However since the calibration includes only half of the delay line X (figure 3.7), the INL of the two-dimensions Vernier TDC grows for higher codes (figure 3.17.a). For the DNL (figure 3.17.b), the folding realized in the two-dimensions Vernier structure produces a sharp peak when consecutive output codes lay on different diagonals. Also in this case the peaks are more evident for higher codes due to the absence of calibration in the last part of the Vernier plane.

The two-dimensions Vernier TDC represents an extension of the standard Vernier TDC. In the two-dimensions approach the number of delay elements required for N quantization levels grows with $\sqrt{N}$ resulting in a set of design constraints that are favorable for TDC with large number of bits. Furthermore, when the two-dimensions Vernier TDC is compared to the standard Vernier TDC, the INL benefits are traded with DNL degradation. This trade-off make the two-dimensions Vernier TDC particularly valuable for systems where the INL is a key parameter.

3.3 TDC gain calibration

In chapter 2 it was shown that the TDC gain calibration is important to have an effective fractional spurs compensation. The TDC calibration system proposed in [21], is capable of background operations and has good accuracy. It is based on a technique that was already proposed for pipelined ADC interstage gain calibration([22]). A deterministic signal is injected in the ADC and the produced digital data is correlated with a digitalized replica of the deterministic pattern, producing a gain error signal. This technique apply very well to a Fractional-N ADPLL, because such a deterministic pattern already exist that is processed by the TDC: the Fractional-N error. A schematic representation of the solution is depicted in figure 3.18. The $error$ signal (the input of the loop filter) is multiplied with the fractional spurs compensation signal. The average of the resulting signal is the correlation of compensation and error signals and therefore contains a gain error information. If the gain error is

completely canceled, the correlation becomes zero and the accumulator holds its value.

The accuracy of the gain calibration depends both on the algorithm and on the gain control accuracy. The accuracy of the algorithm can be improved reducing its speed, while to improve the gain control resolution it is possible to adjust it in the digital domain. To guarantee a high accuracy to the gain adjustments, a hybrid approach has been used: a coarse gain control is performed acting on the TDC delay lines and a fine gain adjustment is performed with a digital multiplication of the TDC output signal. The TDC delay lines calibration acts on the quantization process, and therefore calibrates both the TDC resolution and the full-scale. The accuracy of these parameters is not important, however if their spread from the nominal values are reduced, less margins have to be used in the design to guarantee the absence of TDC saturation (smaller full-scale) or an excess of quantization noise (coarser quantization).

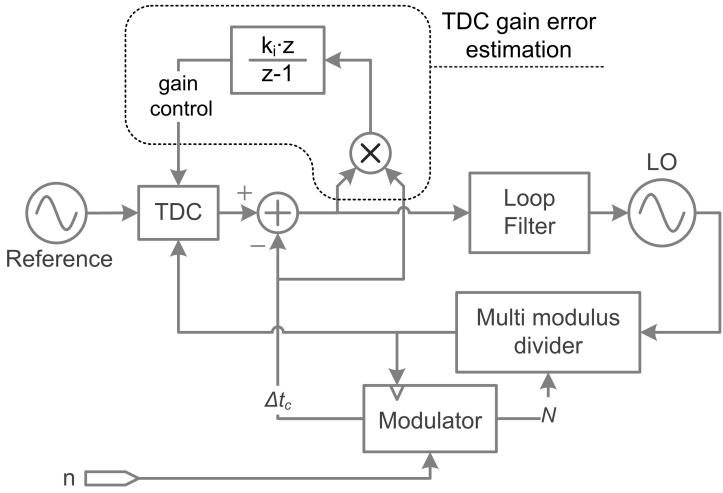

Figure 3.18: TDC gain calibration

3.3.1 Calibration stability

The TDC gain calibration is based on a nonlinear feedback; the nonlinearity is due to the presence of the multiplier in the loop. Therefore it is not simple to guarantee the stability of the loop, because of the absence of a general criterion as in the case of linear loops. The chosen design approach is to study a linearized model and verify the argued conclusions through extensive simulations.

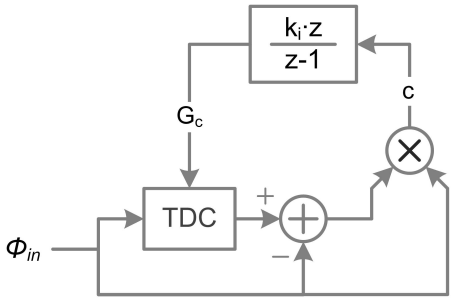

Figure 3.19: Calibration loop phase domain model

Consider the simplified phase domain model of the calibration loop depicted in fig. 3.19. The TDC output ϕ_{TDC} can be expressed as:

$$\phi_{TDC} = \phi_{in} \left(G_n \cdot G_c \right) \tag{3.4}$$

Where G_n and G_c are respectively the natural TDC gain (the quiescent gain value) and the gain calibration coefficient. The output of the multiplier (c) can be therefore expressed as:

$$\begin{aligned} c &= \phi_{in} \left(\phi_{TDC} - \phi_{in} \right) \\ &= \phi_{in}^2 \left(G_n \cdot G_c - 1 \right) \end{aligned} \tag{3.5}$$

In order to linearize the system, ϕ_{in} is assumed to be constant. Even if this assumption is questionable, it allows to understand the effect of the various parameters and to derive some constraints on k_i. In figure 3.20 the resulting

linear model is depicted. The open loop transfer function is a simple accumulator, therefore model bandwidth is directly proportional to the product $G_n \cdot \phi_{in}^2$. If the model bandwidth is much less than the smallest frequency content of ϕ_{in} the former assumption of constant of ϕ_{in} is not violated. Since the system bandwidth depends on ϕ_{in} (assumed constant), it is advisable to use the maximum value of ϕ_{in} in order to not underestimate the bandwidth value.

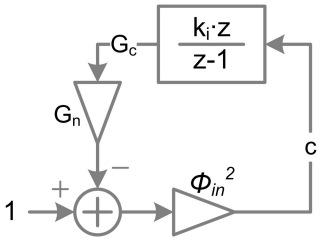

Figure 3.20: Calibration loop linear model

It is remarkable that the underling assumption beneath this calibration algorithm is that the ϕ_{LO} disturbances are uncorrelated to Δt_c. However since the loop is closed and the spurs compensation residue can modulate the LO, this assumption is reasonable only for narrow band ADPLLs. The presence of the ADPLL loop complicates the analysis, but its effects can be taken into account through the behavioral simulations.

Chapter 4

The Digitally Controlled Oscillator

Contents

The Digitally Controlled Oscillator (DCO) is a particular type of Digital to Analog Converter (DAC) where the analog output is the frequency of a Radio Frequency (RF) signal. In chapter 2 it was shown that the DCO represents a key element in the design of a spurs-free and low noise ADPLL.

It does not exist a great variety of DCO architectures, even if different works have been published on the DCO. Most of the DCO proposed in literature are based on passive resonating tank, which major advantage is the excellent purity of the generated signal.

The DCO architecture that was adopted in the ADPLL prototype was proposed in 2010 by Fanori, Liscidini and Castello [23]. The DCO that was used in the prototype was entirely designed by Fanori. The author has not taken part to the DCO ideation and design. However, for completeness, this chapter starts with an overview of the adopted DCO. In the second part of the chapter the DCO predistortion and calibration are presented, which are a part of the work of the author.

4.1 DCO with capacitive degeneration

In an ADPLL, the most critical circuits are the TDC and the DCO. A fine resolution of the TDC is required to minimize the quantization noise introduced in the PLL band, while a tiny frequency discretization of the DCO allows to reduce the noise added far from the carrier. Although the technology evolves (by itself) in the direction of these goals, because of shorter delay stages and smaller parasitic capacitances, the target resolutions for wireless applications are still quite challenging. In the design of a DCO for GSM applications, the target frequency resolution of few kHz, with respect to a tuning range of several hundred MHz around the carrier (e.g. 400MHz in GSM [24]), results in unitary capacitive elements of the order of atto-Farad that cannot be reliably integrated.

A possible solution is the use of capacitive divider networks to obtain a reduction of the minimum effective capacitance that can be switched in parallel to the tank [25]. This approach can improve the DCO frequency resolution, but its sensitivity to mismatches and parasitics limits the robustness of the final design. A more reliable technique [24] consists of the dithering of the least significant bits of the DCO frequency control word (like in a sigma delta DAC). This solution reduces considerably the equivalent DCO frequency resolution (from 12kHz to 30Hz in [24]) but, as it occurs in any sigma-delta data converter, the quantization noise is moved to higher frequencies where the phase noise specs may be even more challenging. Due to this problem, the frequency of dithering must be very high (e.g. 225MHz) to satisfy the emission mask requirements far away from the carrier [24].

All the solutions presented in literature try to improve the DCO resolution working at the level of the oscillator tank, either making a custom design of the capacitive element or exploiting some kind of shrinking effect of the elements of the LC resonator.

The ADPLL prototype uses a DCO architecture based on the capacitive degeneration of an LC-tank oscillator [23]. In this DCO architecture, a portion of the tuning bank is moved from the tank to the sources of the switching pair of the LC oscillator exploiting an intrinsic shrinking effect (figure 4.1). The portion of the capacitive array still in parallel to the inductor (the coarse

tuning bank) is used to compensate process and temperature variation, while the portion at the sources of M_1 and M_2 (named fine tuning bank) is used for the DCO modulation inside the ADPLL.

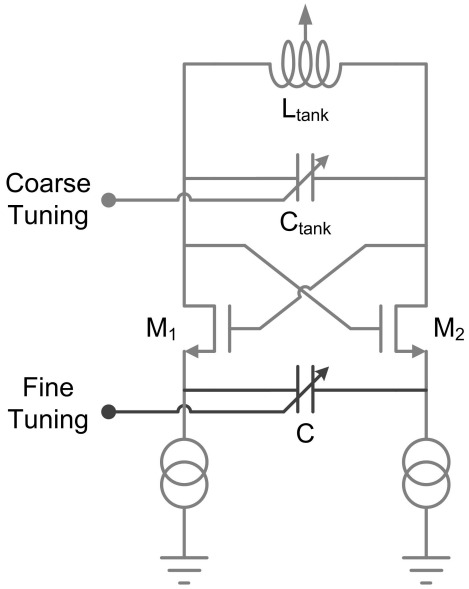

Figure 4.1: Capacitive degeneration LC tank DCO

This capacitance C is reflected in parallel to the tank shrunk by a factor proportional to the transconductance used in the cross-coupled pair. This allows to perform a fine frequency tuning with a resolution that is not limited by the unitary elements present in the capacitor bank.

Since it is very difficult to guarantee a continuity between the fine and coarse tuning characteristics, the fine tuning must have adequate dynamic range in order to keep the ADPLL locked without using the coarse-tuning bank. Moreover the fine tuning dynamic range has to be compliant with the direct modulation constraints. In the GSM transmitter case, considering modulation and thermal drifts, the above condition is satisfied with a fine-tuning range of a few MHz, (800kHz were assumed in [24]).

4.1.1 Capacitive degeneration tuning

It is possible to theoretically derive the tuning characteristic of the oscillator of figure 4.1 ([23]). The tuning characteristic as a function of the capacitance C placed between the sources of the MOS transistors is expressed by the equation

$$f_{DCO}(C) = \sqrt{2}\pi\omega_0\sqrt{1 + Q_f^2\frac{C}{C_{tank}} + \sqrt{4Q_f^2\frac{C}{C_{tank}}}} \tag{4.1}$$

where the parameters Q_f and ω_0 are:

$$Q_f = \frac{gm}{2\omega_0 C} \tag{4.2}$$

$$\omega_0 = \frac{1}{\sqrt{C_{tank}L_{tank}}} \tag{4.3}$$

The DCO fine frequency tuning characteristic is depicted in picture 4.2. It

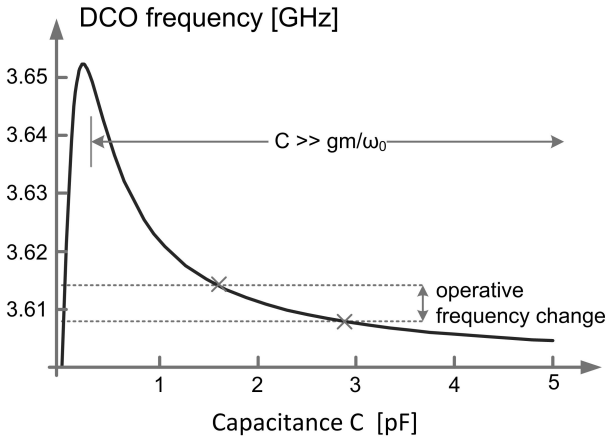

Figure 4.2: DCO fine frequency tuning characteristic

can be noticed that the tuning characteristic is highly nonlinear. To obtain a more linear characteristic the circuit was designed ([23]) to use only the

portion of the curve were $C >> gm/(2\omega_0)$. Under this assumption, equation 4.1 can be simplified to

$$f_{DCO}(C) = 2\pi\omega_0\sqrt{1 + Q_f^2\frac{C}{C_{tank}}} \tag{4.4}$$

Even if equation 4.4 is simpler than equation 4.1, this relationship still remains nonlinear.

4.2 DCO predistortion

Due to it's analog nature, the k_{LO} gain is subject to process and environmental factors that cannot be known precisely. An accurate estimation and correction of k_{LO} is required to apply successfully the two-point modulation scheme (section 2.4). Staszewski in [9], proposes an ADPLL with a normalized DCO. The normalized DCO was made of a digital gain and a DCO. He proposed to calibrate the digital gain with a background algorithm based on the observation of the phase error.

When the modulation signal is wide-band, the DCO fine tune can span a great part of its dynamic range. In section 4.1.1 it has been mentioned that the relationship between frequency and DCO fine tuning is nonlinear; such a nonlinearity distorts the modulation signals producing unwanted components in the synthesized spectra.

In order to mitigate the DCO nonlinear effects, the FCW can be predistorted; basically the same approatch of [9] can be extended to a normalized linearized DCO. However predistortion is computationally non-trivial; if this block is placed in the loop, it will affect a critical signal path. The alternative solution that has been adopted is depicted in figure 4.3. The predistortion is not performed within the loop but on the two-point modulation signal. The main advantage of this solution is to reduce the computational complexity on the most critical path of the system. The accuracy of the resulting two-point modulation is not affected as it is demonstrated in equations 2.37, were the LO parameters are used only to generate the signal that is injected at the LO input.

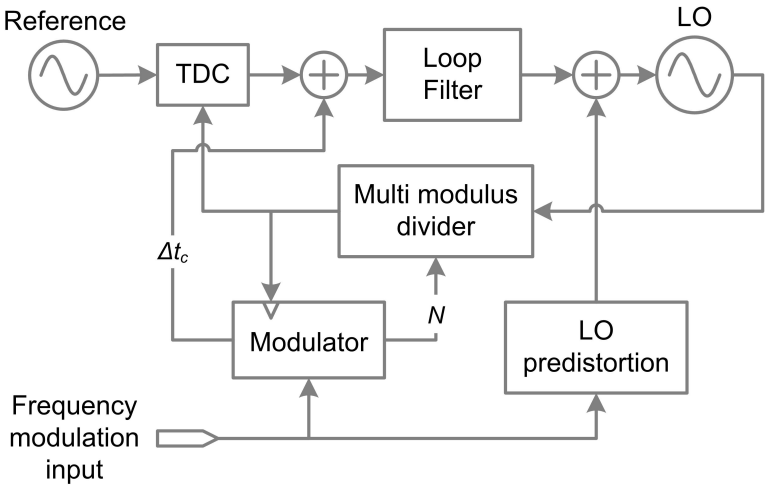

Figure 4.3: Two point modulation with DCO predistortion

However, incorrect values of k_{LO} also affect the accuracy of the loop transfer function, which is not a major concern itself. To guarantee the system stability and a reasonable accuracy of the ADPLL transfer function, a coarse adjustment in the loop filter gain can be performed acting on the filter coefficients without augmenting the loop filter complexity.

A polynomial equation was used for the LO predistortion. Polynomial equations can be used to approximate a nonlinear function as stated by the Taylor polynomial theory. The higher is the order of the polynomial equation, the more accurate is the approximation of the nonlinear function. However, higher polynomial orders augment the implementation complexity. The choice of the polynomial order results in a trade-off between the approximation accuracy and the implementation complexity. A second order polynomial equation was chosen for the LO predistortion.

The LO predistortion second order polynomial equation 4.5, is defined by two parameters k_p and k_q.

$$P(x) = k_p \cdot x + k_q \cdot x^2 \tag{4.5}$$

To evaluate these parameters, a calibration algorithm has been used. The algorithm does not require any complicated maths but the synthesis of particular frequency and therefore is not compatible with background operations. It has to be noticed that this can be a limitation in communication standards where calibration time slots are not available.

4.2.1 DCO predistortion calibration

The idea beyond the adopted calibration algorithm is defining the second order polynomial curve that approximately relates the LO frequency with the LO frequency control word:

$$FCW = f(FMI) \tag{4.6}$$

The parameters of equation 4.5 can be evaluated through the knowledge of three couples of points $[FMI, FCW]$ that belong to the nonlinear curve.

The first steps of calibration are:

- setting to zero both k_q and k_p
- locking the ADPLL to the central frequency of the desired channel acting on the the frequency modulation input (FMI)
- storing the value FMI_0 in O_{FMI}

The FCW variable is therefore settled to a stable value FCW_0; the first of the three points has been found, that is $[FMI_0, FCW_0]$. Once the loop filter output FCW is stored in the O_{FCW} register (fig.4.4), the FMI is changed to an arbitrary value FMI_1.

The loop filter output is then disconnected from the DCO control (which is hold to the O_{FCW} value) and is used to act on the k_p variable. Once the system converge to steady state, and the LO frequency is equal to FMI_1, the LO frequency control word FCW can be expressed as:

$$FCW_1 = O_{FCW} + k_p^{'} \cdot (FMI_1 - O_{FMI}) \tag{4.7}$$

where $k_p^{'}$ is the final value of the loop filter output.

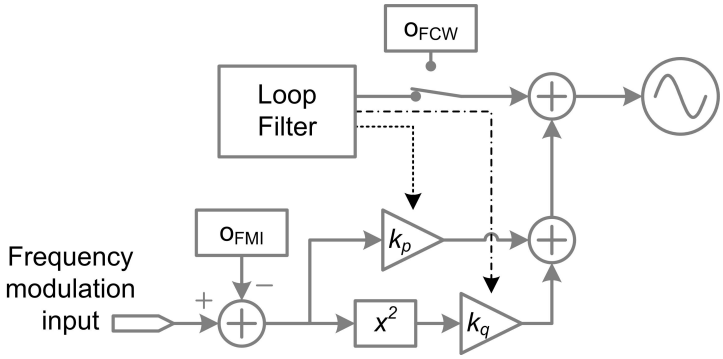

Figure 4.4: Predistortion calibration

At the end of the calibration process a couple of k_q and k_p values will be found that respect equation 4.5 for the point $[FMI_1, FCW_1]$. Therefore from equation 4.7 it is possible to derive:

$$\begin{cases} FCW_1 &= O_{FCW} + k_p^{'} \cdot (FMI_1 - O_{FMI}) \\ FCW_1 &= O_{FCW} + k_p \cdot (FMI_1 - O_{FMI}) + k_q \cdot (FMI_1 - O_{FMI})^2 \end{cases}$$

$$\Rightarrow k_q = \frac{k_p^{'} - k_p}{FMI_1 - O_{FMI}} \tag{4.8}$$

The last step is to set a new FMI value FMI_2, the loop filter output is still used to chainge the k_p, but this time also k_q is corrected applying equation 4.8. Once the system has reached the steady state, the final couple of k_q and k_p values has been found that respects equation 4.5 for the three points $[FMI_0, FCW_0]$, $[FMI_1, FCW_1]$ and $[FMI_2, FCW_2]$.

In figure 4.5 simulation results of LO calibration and modulation are reported, and the different calibration steps are highlighted.

Implementation details

It was claimed that the proposed calibration algorithm does not require any complicated arithmetics; however the presence of a division in 4.8 may seem

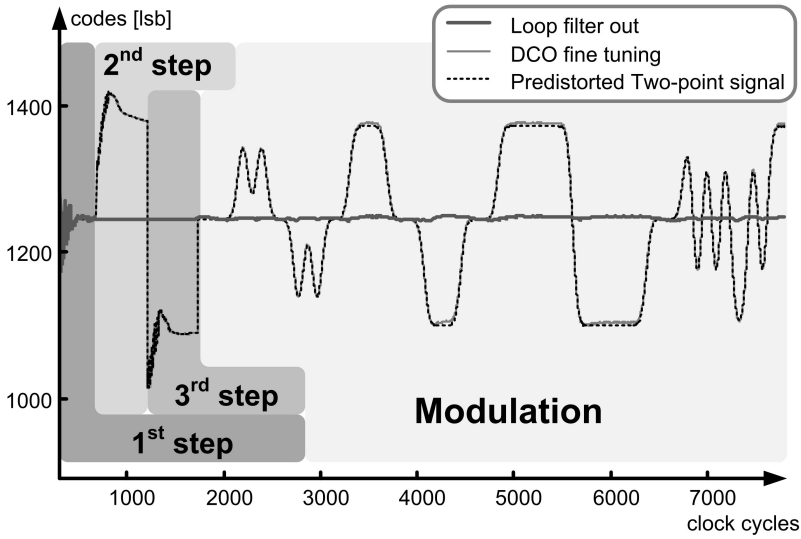

Figure 4.5: Predistortion calibration simulation

a contradiction. However, since the FMI_1 value is arbitrary, it can be chosen such that $(FMI_1 - O_{FMI})$ is a power of two. This trick greatly simplifies the digital hardware implementation.

Another detail of the calibration has been neglected in the previous description. During the locking of the ADPLL to the frequencies FMI_1 and FMI_2, the FCW is calculated with additional operation. The ADPLL stability has to be guaranteed also in this case. In the second step when the target frequency is FMI_1, the loop filter is multiplied by $(FMI_1 - O_{FMI})$ before being added to O_{FCW}. This gain has to be computed in the open loop transfer function. To compensate this effect, the loop filter gain has to be reduced by the same factor. The choice of $(FMI_1 - O_{FMI})$ to be a power of two, is advantageous also in this case.

In the last calibration step the relationship between k_p and FCW becomes:

$$FCW = \acute{k_p} \cdot x_2 + k_p \left(x_2 - \frac{x_2^2}{x_1} \right) \tag{4.9}$$
$$x_1 = FMI_1 - O_{FMI}$$
$$x_2 = FMI_2 - O_{FMI}$$

Choosing x_2 to be equal to $-x_1$.

$$FCW = -\left(\acute{k_p} \cdot x_1 + 2x_1 \cdot k_p \right)$$

Apart from the constant $\acute{k_p} \cdot x_2$, the loop filter is modified by the gain $-2x_1$. Once again the gain is a power of two, and therefore the compensation of its effect (also considering the sign) is simplified.

Chapter 5

ADPLL acquisition of phaselock

Contents

In the previous chapters, it was assumed that the loop was already in lock; but a loop starts out in an unlocked condition and must be brought into lock, either by its own natural actions or with the help of auxiliary circuits. The process of bringing a loop into lock, called **acquisition**, is the subject of this chapter.

An important issue in frequency synthesis for today's wireless applications is the acquisition or settling time to a new channel frequency from the trigger event to the instance when the wireless terminal is ready to transmit or receive with the specified low level of frequency error, phase noise and spurious tones. Loop bandwidth in traditional PLL circuits is fixed to a narrow value to guarantee proper quality of the synthesized clock during the normal operation. Unfortunately, it also severely slows down the loop dynamics.

The poor self-locking capabilities of the proposed ADPLL are initially discussed. Then some auxiliary circuits, that were designed to allow rapid phaselock acquisition, are presented.

5.1 ADPLL aided acquisition

A type two ADPLL contains two integrators, one in the DCO and one in the loop filter. A state variable is associated to each integrator : phase and frequency. To bring the loop into lock, it is necessary to set each of the state variables into close agreement with the corresponding conditions of the input signal. Therefore, a designer must plan for phase acquisition and frequency acquisition [3].

The main limit of the ADPLL self locking is that the TDC does not give any direct frequency information. The TDC estimates the delay between the reference and the divider output edges whatever their frequencies are. Even if multiple signal cycles could occur in one reference period, the TDC would not detect them. Frequency detection is therefore required and this topic will be discussed in the next section.

Furthermore the phase detection of the TDC is not efficient for phaselock acquisition. In previous chapters, it was shown that the proposed ADPLL minimize the TDC full-scale requirements. The designed TDC has a full-scale of $600ps$ that is less than 2% of the reference period ($38ns$). The initial phase error can be as big as the reference period; therefore it is clear that the information that the TDC can provide during phase locking is quite limited. During phase locking, the TDC saturation is a highly nonlinear phenomena, which is highly undesirable. To overcome these limitations a solution will be proposed in next sections: edge searching.

The ADPLL loop filter design has a great impact on phaselock acquisition. Wider loop bandwidth benefits the locking process, but narrow bandwidth benefits the LO spectral purity. The spectral purity of an ADPLL is a property that can be evaluated only for phase locked systems. Therefore it cannot be considered during locking acquisition. The Gear Shifting approach starts the ADPLL with a wide acquisition bandwidth and it narrows it down once locking has been acquired.

All these concepts have been applied to the ADPLL to speed up the phaselock acquisition, resulting in a locking procedure that will be shown in the next sections.

5.2 Frequency Locked Loop

The first step of the locking procedure is to lock the ADPLL frequency to the desired value. Since the TDC does not provide any frequency information, a frequency detector is required to be used in an auxiliary Frequency Locked Loop (FLL). A counter that counts the number of LO periods that occurs in one reference period has been used (fig.5.1). The counter output is subtracted to the division factor, the result (frequency error) is processed by the FLL filter. The frequency detection derivative effect compensates for the LO integration. Therefore the cascade of LO and frequency detector does not perform any integration. The FLL filter is designed as a simple integrator. The resulting system has therefore only one integrator in the loop, resulting unconditionally stable.

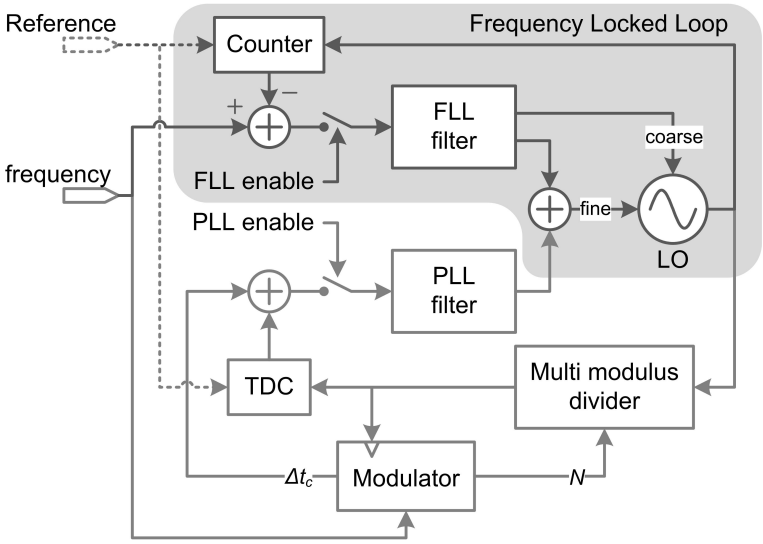

Figure 5.1: ADPLL with frequency loop

Counter quantization effects

The counter is build by a free running counter and a sampler. The free running counter is clocked by the LO and its output continuously wraps around without losing any edge. The free running counter can be modeled in the phase domain simply as a quantizer with resolution of one LO period. The free running counter output is then sampled and held at the reference rate. The sampled data is then differentiated (taking care of handle the wrapping) to obtain a frequency information. The resulting frequency quantization noise is high-pass shaped due to the differentiation process. However, the quantization noise energy is high due to the coarseness of the free running counter quantization and it is typically concentrated in tones.

The counter frequency quantization noise set the main limitation to the FLL bandwidth. The wide FLL bandwidth greatly deteriorates the LO spectral purity, and can also have negative effects on the phase lock acquisition. The narrow FLL bandwidth produces precise frequency locking benefiting the reliability of the phase lock acquisition. The narrow band FLL slows down the frequency lock acquisition, but as will be shown in the next sections it has a moderate impact to the total locking time.

5.3 Edge search

Once the FLL has reached steady state, the phaselock has to be acquired. It was already mentioned that the tiny TDC full-scale does not allow a reliable phaselock acquisition. With high probability, the TDC output will be deeply saturated and therefore the activation of the phase loop would lead to a highly nonlinear system. In this scenario, the resulting dynamics of system variables would be unpredictable and even the loop stability can not be guaranteed.

Consider an example where the target LO frequency is six times greater than the reference frequency, and the FLL has reached steady state (figure 5.2). The LO frequency is almost correct but the delay between the divider output and the reference is grater than the TDC full-scale. The TDC output is therefore saturated. However, for each reference rising edge, a LO rising edge exists whose delay is less than the TDC full-scale. The problem is that the divider

has a state variable whose initial condition is unknown.

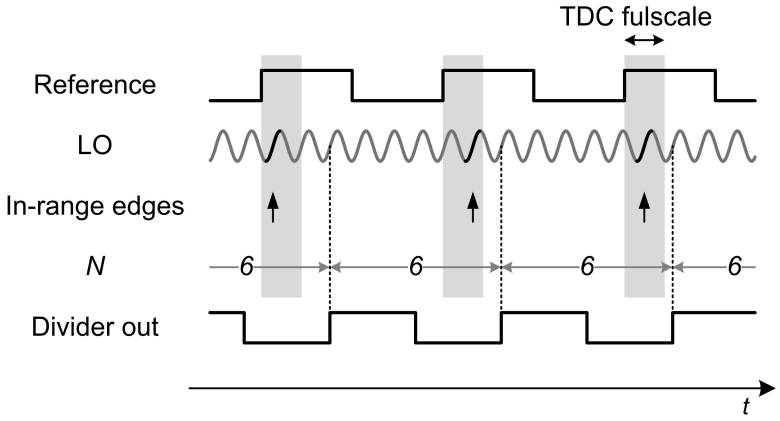

Figure 5.2: Time diagram before edge searching

One possible solution is to reset the divider state variable synchronously with the reference positive edge. Then the PLL can be enabled (and the FLL disabled). In this way the delay between the reference and the divider positive edges will be greatly reduced. The major drawback of this solution is that the divider is synchronous to the LO (and therefore asynchronous with the reference clock) and that the LO frequency is high. This use of an asynchronous reset demands that once the reset is disabled the divider operation starts immediately. This will produce severe timing constraints to the design of the multi modulus divider, which itself is a nontrivial task.

The solution that was adopted is to act on the multi modulus divider control, trying to push the divider output edges within the TDC full-scale. Acting on the multi modulus divider control, only the divider state variable is manipulated without affecting the frequency loop that is kept active. In practice if the TDC output is upper saturated the divider control is reduced, while if TDC output is lower saturated the divider control is increased (figure 5.3).

A schematic representation of the proposed system is depicted in figure 5.4. The edge searching can be classified as a type one phase locked loop: the TDC still acts as a phase detector while the divider acts as a local oscillator. As-

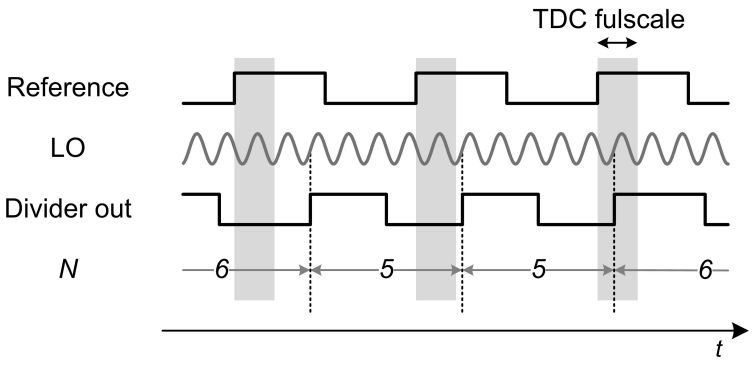

Figure 5.3: Edge searching time diagram

suming the LO frequency to be a constant value the divider output frequency is inversely proportional to the divider control word (with proportional coefficient equal to the LO frequency), therefore the divider can be modeled as a controlled oscillator. The great advantage of this phase locked loop is that starts with a very small frequency error.

In figure 5.4 it is highlighted that the edge search loop and the FLL are not linked; therefore their stability can be studied independently. Since the edge search operations are performed with a saturated TDC, linear tools do not apply. However the presence of a single integrator in the edge search loop greatly simplifies the system design and verification.

This solution performs a linear search of the edge. A possible improvement could be a binary search. In such a case the multi modulus divider control would not be incremented or decremented only by one. The idea of the binary search is that if the TDC is lower saturated at the beginning of the procedure, the divider control is increased of half of its value. If the TDC is still saturated, the divider control is increased or decreased of one forth, dependently of the saturation sign. The method successively refines the search, so it finds the rising edge in a logarithmic number of iterations. It must be taken into account that real multi modulus divider have a limited range of allowed control words, therefore the designer must verify that the binary search does not exceed that

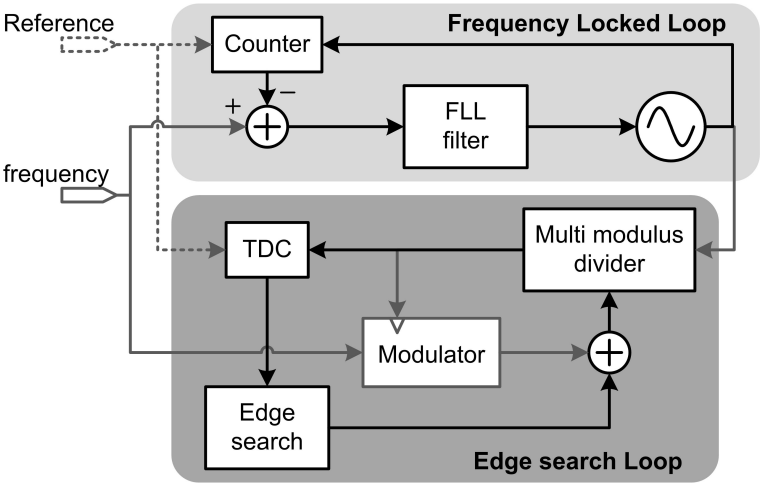

Figure 5.4: ADPLL with edge searching

range. This solution has not be included in the prototype, but it does not seems to have significant drawbacks.

Falling edge issue

It must be noticed that the edge search algorithm has two equilibrium points: one stable and one instable. The stable one is associated with the presence of a divider output rising edge in the TDC range, the unstable one is a divider output falling edge in the TDC range. If initially the system is in the unstable point the search process would not start. Even if the system would naturally move from this equilibrium, this process can be slow and unreliable. To overcome this practical limitation, a control logic was added. When the system enters for the first time in an equilibrium point, it is forced to move from it still modifying the divider control. Then the edge search is restored, bringing the system to the stable equilibrium point. If it was entered in an unstable point it has been forced to move from it; otherwise the system has been perturbed but it has finally returned to the same stable equilibrium point.

5.4 Gear Shift

During the normal course of PLL operation, two unique phases with conflicting requirements are readily distinguished. In the first interval, the goal is to acquire the desired frequency as soon as practically possible; deteriorated LO spectral purity is tolerated at that time. In the second interval, which covers the actual transmitter (TX) and receiver (RX) operations, the goal is to maintain or track the desired frequency that was acquired during the first phase, with higher spectral purity.

The gear shift technique addresses this issue with the use of linear time-variant loop filter. The approach is to start with the ultra-wide acquisition bandwidth and to narrow it down once the locking is reached. This technique has been propose for analog PLL ([26]). However it is generally difficult to perform a PLL gear shifting in analog circuits because of the imperfect matching and voltage or charge losses during switching. Due to the fully digital implementation, the manipulation of filter coefficients and state variables is very simple.

The loop filter is supposed to be an infinite impulse response (IIR) filter[1], thus it can be described and implemented in terms of the difference equation that defines how the output signal is related to the input signal:

$$y[n] = \frac{1}{a_0}\left(\sum_{i=0}^{P} b_i x[n-i] - \sum_{j=0}^{Q} a_j y[n-j]\right) \tag{5.1}$$

When gear shift is performed, the loop filter coefficients are changed to the values of the new filter transfer function. This action will produce variations in the filter output that have to be minimized. Output variations depend on the state variables. Different criteria can be applied to manipulate the state variables. One possibility is to act on state variables forcing all the derivative initial condition null. Alternatively, the derivative terms can be reduced proportionally to the ratio between initial an final bandwidth. A rigorous comparison between gear shift strategies as not been developed yet.

[1] A type two ADPLL requires a pole in DC and this cannot be mapped in a finite impulse response (FIR) filter

In the prototype the zero derivative strategy was implemented. The system design was based on simulations performed on a behavioral time domain ADPLL model.

The gear shift was applied not only on the initial phaselock acquisition, but also during the LO predistortion calibration (section 4.2.1) to accelerate the locking of the system to the calibration points.

5.5 Lock acquisition procedure

All the acquisition aiding circuits that were shown in this chapter are sequentially activated during the acquisition procedure. A dedicated finite state machine controls the acquisition process, based on feedback given by the acquisition aiding circuits. In figure 5.5 the results of a simulation of the ADPLL behavioral time domain model are depicted.

The locking procedure starts with the acquisition of frequency locking: the FLL is activated and the DCO coarse tuning settles to its steady state value. When the FLL is activated also the DCO fine tuning is controlled but its value is always kept near to the center of its dynamic range. The "phase error" depicted in figure 5.5 is the phase difference between the reference and the divider output, without any quantization or saturation; this variable does not exist in the ADPLL, but the TDC output is its digitalized replica. While the coarse tuning is settled to its final value, the phase error derivative approaches zero while its absolute value can be arbitrary high.

The second stage of acquisition is the edge search. Looking at figure 5.5 it is possible to see that during edge search the divider control sequence is modified and the phase error settles to zero. The phase error settling is slew-rate limited; this is an effect of the linear search method (as explained in section 5.3). During the edge search the FLL is still active and continuously updates coarse and fine DCO tuning. The large fine DCO tuning signal amplitude is due to the large FLL quantization noise.

Once the edge search is completed, the PLL is activated in wide-band mode. The phase lock is achieved as can be seen from spurs correction: the error (the difference between the TDC output and the correction sequence) settles to zero. The coarse tuning value is frozen and only the DCO fine tuning is

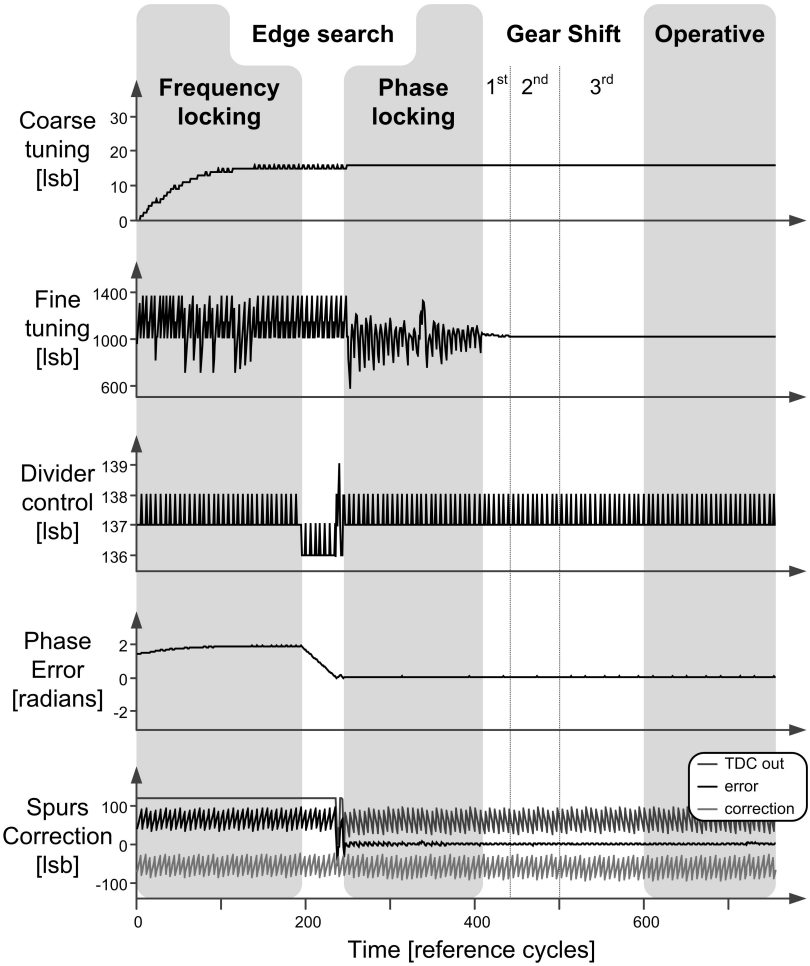

Figure 5.5: ADPLL locking procedure simulation

controlled by the PLL. The large fine tuning signal amplitude is due to the wide-band PLL mode.

The PLL is successfully locked but it is still in wide-band mode. The Gear

shift is then performed to bring the PLL to narrow-band. The simulation refers to a multiple Gear shift performed in three steps. The fine tuning signal calms down. The absence of spikes in the spurs compensation error highlights the effectiveness of the gear shift.

Once the gear shift is finished, the PLL is in an operative state and the DCO spectral purity is optimal.

Chapter 6

Prototypes

Contents

The research activity presented in this dissertation has produced two prototypes: the two-dimensions Vernier TDC and the ADPLL test chips. The TDC prototype was mainly designed and characterized by the author, in collaboration with the Italian design center of Marvell, under the supervision of Fernando De Bernardinis (Marvell), Antonio Liscidini and Rinaldo Castello (both from the University of Pavia). The measurements results were presented to the 2009 IEEE Custom Integrated Circuits Conference ([19]) and published on IEEE Journal of Solid-State Circuits ([20]).

The ADPLL prototype was designed in collaboration with the Italian design center of Marvell. The ADPLL design has involved more people both form the University of Pavia and from Marvell. The team was mainly formed by the author, Luca Fanori, Antonio Liscidini (University of Pavia) and Fernando De Bernardinis (Marvell) under the supervision of Francesco Rezzi (Marvell) and Rinaldo Castello (University of Pavia). Also other people form Marvell were involved in particular steps of the project: Danilo Gerna, Luca Romanò, Antonio Milani, Enrico Sacchi, Paolo Rossi, Pierandrea Savo, Valeria Garofalo, Francesco De Paola, Patrizia Mastromatteo and Riccardo Straus.

In this chapter the two prototypes are presented and measurements results are discussed.

6.1 Two-dimensions TDC prototype

In chapter 3 the Two-dimensions TDC topology was introduced, and the circuit elements were discussed. The designed prototype was fabricated in 65nm Low Power standard CMOS technology by TSMC, its most relevant parameters are:

Number of bits	7
Reference frequency	$50MHz$
Time resolution	$5ps$
Full scale	$590ps$

Table 6.1: TDC prototype parameters

In figure 6.1 the block scheme of the TDC prototype is depicted. The test chip comprises the two-dimensions Vernier TDC core circuits, the input network, the relative calibration digital logic, the thermometric-to-binary decoder and an I^{2}C slave unit. The I^{2}C unit was used to externally access configuration registers.

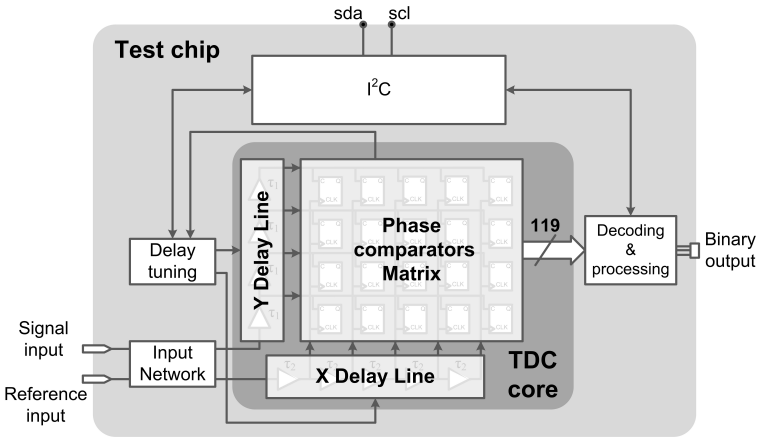

Figure 6.1: TDC prototype scheme

The design of the prototype was performed with a full custom approach to the TDC core and the input network; while a VHDL[1] based digital design flow was used for the relative calibration digital logic and an I^2C unit. Also for the physical layout two different approaches was used: a full custom layout for the TDC core and the input network, and an automated place and route of standard cells for the digital circuits.

The obtained chip is depicted in figure 6.2. It is possible to see that the prototype layout is highly pad limited, and the active area ($280\mu m \cdot 280\mu m$) is a small fraction of the total area.

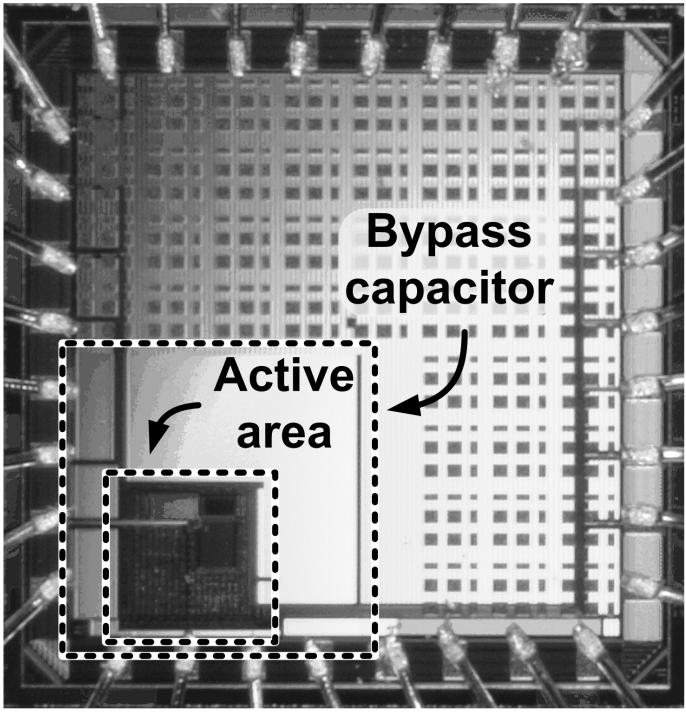

Figure 6.2: TDC prototype photograph

[1] VHDL: VHSIC Hardware Description Language, VHSIC: Very-High-Speed Integrated Circuit

6.1.1 Measurement results

The characterization of the prototype was performed on three main parameters:

- TDC resolution
- TDC linearity (INL and DNL [2])
- Power consumption

The TDC Test Bench (TB) is outlined in figure 6.3. The TB is based on a Data Timing Generator (DTG, the specific instrument is the Tektronix DTG5274) that is able to generate square waves with edges slope up to $5GV/s$ and to set precisely the delay between the signals. Through a GPIB [3] connection between the DTG and a personal computer, various tests were performed varing the clock-signal delay. Since the DTG requires to switch off the output signals to change their delay, the measurements were performed in a step mode. Tests were performed with a delay sweep of 900 points with 1ps steps. For each step multiple samples were acquired (up to 10000).

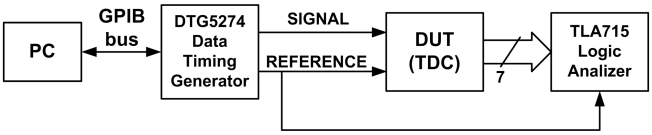

Figure 6.3: TDC test bench

INL and DNL were calculated with a histogram method ([1]), the first linearity measurements results are represented in figure 6.4. The DNL is always less the one LSB while the INL shows a maximum of 3.3 LSBs. The big bump present in the INL measure is incompatible with the folding existing in the proposed architecture, which demands a periodicity in the INL (as explained in chapter 3). In order to identify the cause of this unexpected behavior

[2] INL: Integral Non Linearity, DNL: Differential Non Linearity ([1])
[3] GPIB: General Purpose Interface Bus, is an 8-bit bus standard (IEEE-488)

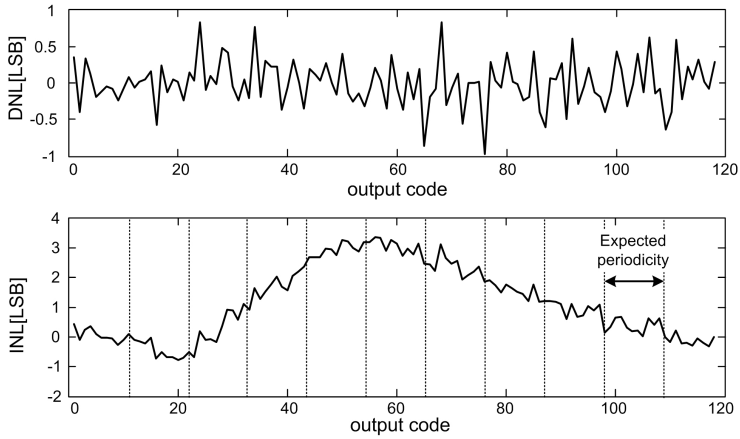

Figure 6.4: Linearity measurements

the possible sources of distortion have been investigated. Three different distortion sources were analyzed: the comparators, the delay lines and the input network. Unmatched delay elements or unmatched comparators produce an effect on the absolute INL (and DNL) that depends on the time resolution or the offset of the converter. On the contrary, any distortion produced before the quantization process (i.e. within the input network), should have an effect on the absolute INL independent from the value of the TDC time resolution. For this reason, the TDC linearity has been measured varying the time resolution from 4.8ps to 7.9ps (trimming the delay of the Delay line X it is possible to act on the TDC resolution as shown in chapter 3).

The results are reported in figure 6.5. The invariance of the absolute INL bump with the time resolution suggested that the main source of the unexpected distortion occurs before the quantization process (i.e. on the board or most probably in the input network). One assumption is that when the rising edges of the input signals are very closed to each other, a cross-talk in the input network generates a pulling that distorts the delay ramp used to characterize the TDC. The power consumption of the whole TDC has been evaluated for a 50Msps sample rate including calibration phase contribution. Figure 6.6

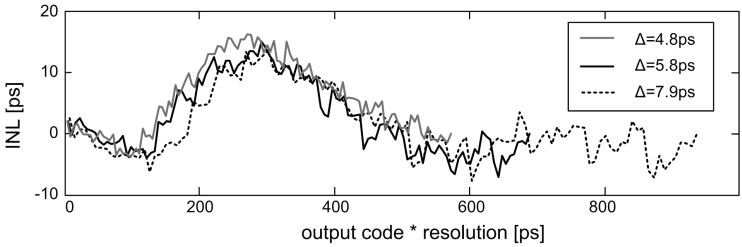

Figure 6.5: Linearity measurements vs TDC resolution

shows the power consumption for different number of capacitance connected to the delay elements of the X delay line (which corresponds to different time resolutions).

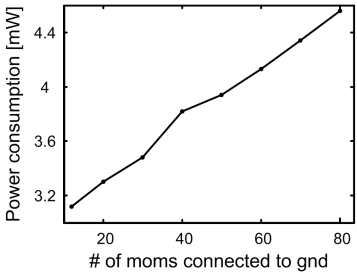

Figure 6.6: Power consumption versus delay trimming

6.2 ADPLL prototype

The ADPLL prototype has been fabricated in 55nm Low Power standard CMOS technology by TSMC. The 55nm process is just a 10% optical shrink of the 65nm process used for the TDC prototype. Figure 6.7 shows a layout snapshot of the prototype; the TDC, the DCO and the digital logic are highlighted in the picture. The chip size is of $2,2 \cdot 1,6mm^2$.

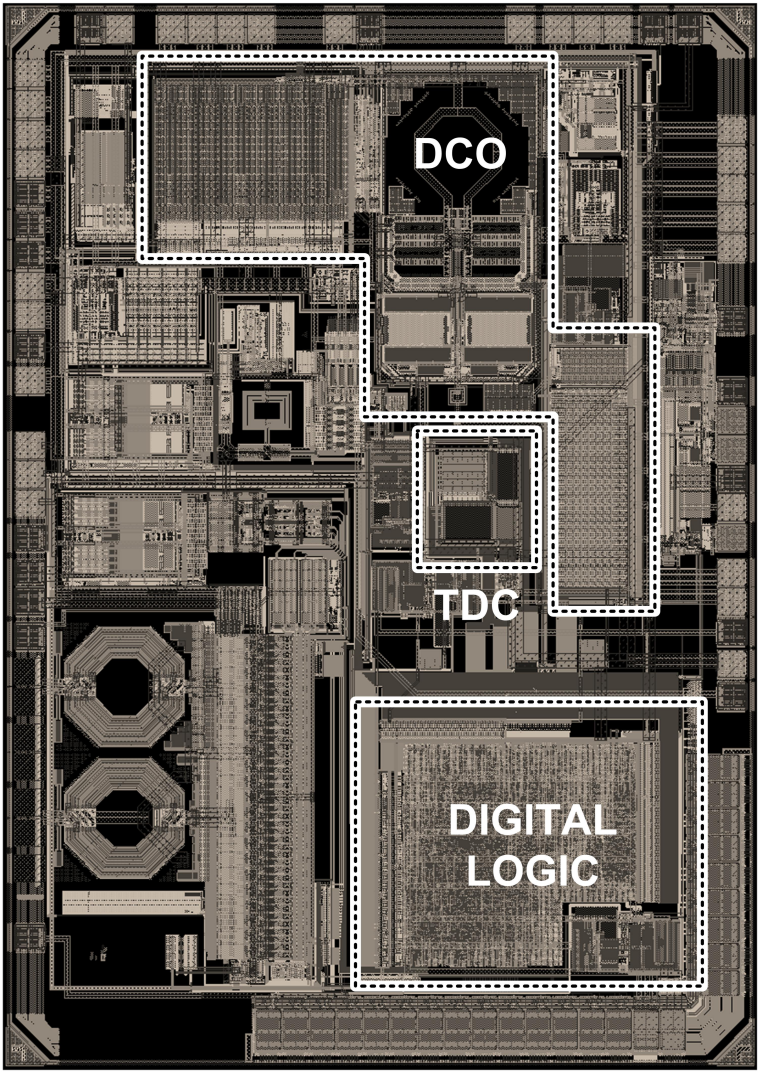

Figure 6.7: ADPLL prototype layout snapshot

The ADPLL comprises a scaled copy of the two-dimensions Vernier TDC designed for the first prototype. The TDC specification are therefore the same of table 6.1.

Most of the circuits (divider, regulators, reference oscillator buffer, ..) were taken from Marvell libraries. The DCO designed by Luca Fanori is a modified version of a VCO previously designed by Marvell. The major DCO specifications are:

Central frequency	$7,2GHz$
Maximum frequency	$8,6GHz$
Minimum frequency	$5,8GHz$
Coarse Tuning Number of bits[1]	$7+6$
Coarse Tuning full scale	$2.8GHz$
Coarse Tuning resolution	$1MHz$
Fine Tuning Number of bits	13
Fine Tuning full scale	$10MHz$
Fine Tuning resolution	$1kHz$

Table 6.2: DCO parameters

Unfortunately the ADPLL prototype characterization is now still in progress, and measurement results are not available at the moment.

[1] The Coarse Tuning is organized in two independent banks. The equivalent number of bit ($log_2\left(fullscale/resolution\right)$) is smaller than $7+6$ due to the overlap between the two independent banks.

Conclusions

The Phase Locked Loop (PLL) design has been a research topic for almost a century (the earliest description of a system similar to a PLL was provided in 1932 [4]). The reasons for this longevity are basically two: the first is that the great flexibility of PLL make it attractive in various systems, the second is that the inherent time variant and nonlinear nature of the PLL make it resistant to theoretical approaches. The All Digital PLL (ADPLL) is the last step in the PLL evolution and is the result of a digital intensive approach to the PLL design.

The main advantage of the ADPLLs is their remarkable reconfigurability. Moreover the availability of digital foreground and background calibration algorithms contribute to their robustness and reliability. One of the most critical aspect of industrial electronic devices is their testing, which is generally particularly complex and expensive for analog devices. The extensive presence of digital signals within the ADPLL system make it easier to test. Furthermore auxiliary digital circuits can be designed to apply Built in Self Test procedures to the analog components ([27]). The key concept of those solution is to design a system able to self test itself, making the device testing cheaper.

Those characteristics make the ADPLL attractive for real word applications and emerging systems such as Software Defined Radios ([27]). The research activity that is described in this dissertation was targeted on the digital intensive PLL design, with the objective to determine solutions that allow to simplify all system analog design constraints.

The TDC has a great impact on the ADPLL performances. In the first part of my activity I was committed to the design of a Time to Digital Converter (TDC). The design of the TDC was addressed without preconceived ideas, resulting in the two-dimensions Vernier TDC architecture. The two-dimensions Vernier TDC represents the most extensive implementation of the Vernier

principle[4]. The two-dimensions Vernier TDC capacities was demonstrated through both a theoretical approach and a prototype characterization.

In the second part of my activity I engaged the ADPLL system design. The ADPLL architecture that resulted from my activity is not highly innovative: it relies on the fractional-N architecture, that is a widespread solution to the PLL design. However, the system design was motivated, highlighting the advantages of the proposed architecture. It was shown that the proposed architecture is characterized by relaxed constraints on the TDC design. Furthermore the extensive use of digital logic greatly simplifies the system design and the technology scaling. Every aspect of the ADPLL was considered: the locking acquisition, the steady state performances and the overall system reliability.

An ADPLL prototype was finally realized, which characterization is now still in progress. The characterization results will hopefully allow to drew further conclusions on the ADPLL potentiality, and on the effectiveness of the adopted digital intensive design approach.

[4] In chapter 3 it was shown that the standard Vernier TDC is "included" in the two-dimensions structure.

Bibliography

[1] F. Maloberti, "Data converters", Springer (2007), ISBN: 978-0-387-32485-2

[2] William C. Lindsey and Chak Ming Chie., "A survey of digital phase-locked loops", Proceedings of the IEEE (1981) vol. 69 (4) pp. 410 - 431

[3] Floyd M. Gardner, "Phaselock Techniques (3rd ed.)", John Wiley & Sons (2005), ISBN 0-471-43063-3

[4] Thomas H. Lee, "The design of CMOS Radio-Frequency Integrated Circuits (2nd ed.)", Cambridge University Press (2004), ISBN 0-521-83539-9

[5] William F. Egan, "Frequency Synthesis by Phase Lock (2nd ed.)", John Wiley & Sons (2000), ISBN 0-471-32104-4

[6] A. Hajimiri and T.H. Lee, "A General Theory of Phase Noise in Electrical Oscillators", IEEE Journal of Solid-State Circuits (1998) vol. 33 (2) pp. 179 - 194

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

[8] T.H. Lee and A. Hajimiri, "Oscillator phase noise: a tutorial", IEEE Journal of Solid-State Circuits (2000) vol. 35 (3) pp. 326 - 336

[9] R. B. Staszewsi, P. T. Balsara, "All-Digital Frequency Synthesizer in deep-submicron CMOS", Jhon Wiley & Sons (2006), ISBN: 0-471-77255-0

[10] Liangge Xu Lindfors et al., "A 2.4-GHz Low-Power All-Digital Phase-Locked Loop", IEEE Journal of Solid-State Circuits (2010) vol. 45 (8) pp. 1513 - 1521

[11] Robert M. Gray, "Oversampled Sigma-Delta Modulation", IEEE Transactions on Communications (1987) vol. 35 (5) pp. 481 - 489

[12] Robert M. Gray, "Quantization noise spectra"', IEEE Transactions on Information Theory (1990) vol. 36 (6) pp. 1220 - 1244

[13] W. Chou, P.W. Wong, R.M. Gray, "Multistage sigma-delta modulation", IEEE Transactions on Information Theory (1989) vol. 35 (4) pp. 784 - 796

[14] K.J.Wang , A. Swaminathan, I. Galton, "Spurious Tone Suppression Techniques Applied to a Wide-Bandwidth 2.4 GHz Fractional-N PLL", IEEE Journal of Solid-State Circuits (2008) vol. 43 (12) pp. 2787 - 2797

[15] J. Borremans et al., "A 86 MHz - 12 GHz Digital-Intensive PLL for Software-Defined Radios, Using a 6 fJ/Step TDC in 40 nm Digital CMOS", IEEE Journal of Solid-State Circuits (2010) vol. 45 (10) pp. 2116 - 2129

[16] R. B. Staszewski et al., "All-digital PLL and Transmitter for Mobile Phones", IEEE Journal of Solid-State Circuits (2005) vol. 40 (12) pp.2469 - 2482

[17] R. Tonietto et al., "A 3MHz Bandwidth Low Noise RF All Digital PLL with 12ps Resolution Time to Digital Converter", Proceedings of the 32th European Solid-State Circuits Conference (ESSCIRC 2006) pp.150 - 153

[18] E. Temporiti et al., "A 3 GHz Fractional All-Digital PLL With a 1.8 MHz Bandwidth Implementing Spur Reduction Techniques", IEEE Journal of Solid-State Circuits (2009) vol. 44 (3) pp.824 - 834

[19] A. Liscidini, L. Vercesi and R. Castello, "Time to digital converter based on a 2-dimensions Vernier architecture", IEEE Custom Integrated Circuits Conference, 13-16 Sept. 2009, San Jose, California, pp. 45 - 48

[20] L. Vercesi, A. Liscidini and R. Castello, "Two-Dimensions Vernier Time-to-Digital Converter", IEEE Journal of Solid-State Circuits (2010) vol. 45 (8) pp. 1504 - 1512

[21] C. M. Hsu, M. Z. Straayer, M. H. Perrott, "A Low-Noise Wide-BW 3.6-GHz Digital $\Sigma\Delta$ Fractional-N Frequency Synthesizer With a Noise-Shaping Time-to-Digital Converter and Quantization Noise Cancellation" IEEE Journal of Solid-State Circuits (2008) vol. 43 (12) pp. 2776 - 2786

[22] E. Siragusa, I. Galton, "A digitally enhanced 1.8-V 15-bit 40-MSample/s CMOS pipelined ADC", IEEE Journal of Solid-State Circuits (2004) vol. 39 (12) pp. 2126 - 2138

[23] L. Fanori, A. Liscidini and R. Castello, "Capacitive Degeneration in LC-Tank Oscillator for DCO Fine-Frequency Tuning", IEEE Journal of Solid-State Circuits (jet to be published but available at *http://ieeexplore.ieee.org*)

[24] R. B. Staszewski et al., "A digitally controlled oscillator in a 90 nm digital CMOS process for mobile phones" IEEE Journal of Solid-State Circuits (2005) vol. 40 (11) pp. 2203 - 2211.

[25] Chen et al., "A 9 GHz dual-mode digitally controlled oscillator for GSM/UMTS transceivers in 65 nm CMOS", IEEE Asian Solid-State Circuits Conference (2007) pp. 432 - 435

[26] Gyoung-Tae Roh, Yong Hoon Lee, and Beomsup Kim, "Optimum phase-acquisition technique for charge-pump PLL", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing (1997) vol. 44 (9) pp. 729 - 740.

[27] R. Staszewski et al., "Software Assisted Digital RF Processor (DRP^{TM}) for Single-Chip GSM Radio in 90 nm CMOS", IEEE Journal of Solid-State Circuits (2010) vol. 45 (2) pp. 276 - 288.

Acknowledgements

I would like to thank everyone who has the determination to get through this dissertation, although my prose is not always thrilling. Everybody else I have to thank are Italian and therefore I will now regain my mother tongue.

Negli ultimi anni ho avuto la fortuna di poter seguire le mie aspirazioni. Per molte delle opportunità che mi sono state offerte, devo ringraziare il professor Castello. Il supporto che mi ha dato va ben oltre a quello legato al suo ruolo: i suoi consigli mi hanno permesso di lavorare con consapevolezza e indipendenza.

Se almeno un poco sono riuscito a limare certi spigoli del mio carattere, è certamente grazie al quotidiano confronto con Antonio, la serenità degli accesi scambi di idee mi ha incredibilmente aiutato a rendere concreto il mio lavoro. Un ringraziamento speciale va a tutti i ragazzi che hanno vissuto con me i momenti più divertenti tra quelli trascorsi al Microlab (e a Otranto). Tra questi, in ordine di apparizione, Roberto e Cesare, Flavio e Masta, le vecchie compagne di corso Marika e Raffaella, e infine Alberto, Luca e Marco. In particolare voglio ringraziare Alex con il quale ho condiviso spazi di lavoro e intervalli (e la camera di Otranto).

Negli ultimi due anni ho trascorso buona parte del mio tempo in Marvell. Lì ho avuto la possibilità di vivere l'industria e di confrontarmi con grandi professionisti che mi hanno accolto in un ambiente molto stimolante. Per questa grande opportunità devo ringraziare soprattutto Francesco Rezzi. Per la pazienza e per la perseveranza che ha dimostrato nello spronarmi a superare i miei limiti voglio ringraziare Fernando. In vero devo ringraziare in pratica tutti in Marvell, in questi due anni quello che ho imparato, l'ho imparato per buona parte dalle loro esaurienti risposte alle mie numerose domande. Ho

davvero apprezzato la loro disponibilità e generosità nel condividere la loro esperienza con me.

Oltre a lavorare in Marvell mi sono anche rilassato e divertito, e spesso lo ho fatto in compagnia di Enrico, Chiara, Francesco, Luca, Valeria, Marika, Flavio, Federico e Stefano; a voi tutti uno speciale ringraziamento.

Per il supporto che mi hanno dato con l'attento lavoro di revisione ringrazio vivamente Roberta e Luca. Questa tesi sarebbe stata decisamente più difficile da leggere se Chiara non avesse messo numerose pezze al mio inglese, per questo la ringrazio.

Non ho ancora ringraziato alcune delle persone più importanti per me. Per l'incondizionato supporto, la fiducia e l'amore ringrazio Alessandra. Forse è scontato ringraziare i miei genitori, ma loro per me sono veramente un riferimento e il loro entusiasmo nell'appoggiare i miei progetti (anche quelli meno promettenti) è stato sempre grande. Grazie a Teresa, Ivano e Tommaso per la splendida accoglienza.

Gli amici di sempre sono una costante e dopo essere già stati ringraziati nelle due tesi precedenti credo siano ormai stufi. Festeggeremo e vi ringrazierò. Infine vorrei ringraziare il basket pavese, e il CB per avermi adottato e per aver apprezzato le mie doti tecniche.

Di qualcuno certo mi sarò dimenticato, non me ne voglia. A mia discolpa va il fatto che i ringraziamenti non si rileggono: è difficile scriverli (esprimere per iscritto i sentimenti non è dote tipica di un ingegnere) e li correggerei senza sosta.

A voi tutti ancora, grazie

Luca

November 15, 2010