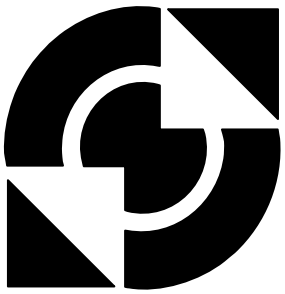
University of Twente

Faculty of Electrical Engineering,
Mathematics & Computer Science



Design of clock cleaner
A fast locking PLL

H.T. Griffioen
Doc. Verslag
August 2008

Supervisors:
prof. ir. A.J.M. van Tuijl
prof. dr. ir. B. Nauta
ir. P.F.J. Geraedts

Report number: 067.3269
Chair of Integrated Circuit Design
Faculty of Electrical Engineering,
Mathematics & Computer Science
University of Twente
P. O. Box 217
7500 AE Enschede

Abstract

Today's communication systems make use of a variety of phase-locked loops (PLL), for instance in burst wise digital audio. Clock/Data signals can be heavily distorted by jitter. Typically PLL's are used to suppress the jitter through their low loop bandwidth, but bring along long settling times as well. In comparison with the amount data to be sent, this settling time can become significantly large and not very power-efficient.

In order to reduce this overhead, a new PLL has been developed. This PLL contains a frequency estimator which estimates the frequency within one period of the incoming clock signal.

A switched capacitor relaxation oscillator has been used in order to integrate the estimator with the VCO of the PLL. This avoided the need of calibration of those two building blocks.

The results are a PLL which can lock within 4 clock periods of the incoming clock. At $6.67MHz$ this is equal to 600ns, which is remarkably fast. It is expected that this can even be faster with only one clock period.

Contents

Chapter 1

Introduction

Today's communication systems make use of a variety of phase-locked loops (PLL). An example is the area of clock and data recovery (CDR), in which PLL's are frequently applied. In this area incoming clock/data signals can be heavily distorted by jitter, for instance due to cross-talk of neighboring wires. A typical way to filter out this jitter is to apply a PLL with a low loop bandwidth. The large time constants in such a system mean a long settling time as well though. During this settling time the clock/data signals can not be received reliably. In applications where clock/data is sent continuously, the overhead of this settling time is negligible, i.e. the PLL has to settle only once at initialization of the system. In applications where clock/data is sent in bursts to save power, this is not negligible cannot be done: at the beginning of every burst the PLL has to settle before clock/data signals can be received reliably. The overhead of PLL settling in such burst-mode systems can be considerable. Decreasing this overhead enables very power-efficient communication systems. A possible solution is combining a low-bandwidth PLL with initial frequency estimation. This project investigates the feasibility of such a solution.

The project is a continuation of the research of prof. ir. A.J.M. van Tuijl and ir. P.F.J. Geraedts who have been working on an idea of prof. M.J. Underhill: *the anti-jitter circuit topology* [3]. A relaxation oscillator with a very good FoM was the result.

Van Tuijl had an idea to shorten the settle time of a PLL [1]. It is called the Clock Cleaner Circuit (CCC). Working on this subject Geraedts has developed a switched-capacitor relaxation oscillator which could be used for this purpose. This continuation of the project involves several questions.

Within the context of the ideas of Van Tuijl several questions are of main interest:

1. What problems will be encountered to realize such a circuit?

2. How could the feedback loop be realized?

3. How to assure long term stability of the oscillator?

1.1 Phase-locked loop background

A typical PLL consists of a phase detector (PD) and a voltage controlled oscillator (VCO) (see Figure 1.1). The output of the VCO is then compared with the input signal. In fact only the two phases ϕ_{in} and ϕ_{out} are being compared. The resulting excess phase $\phi_e = \phi_{out} - \phi_{in}$ will be reduced till zero. The loop is called to be in phase lock as ϕ_e is sufficiently small and constant in time. This means that

$$\frac{d\phi_e}{dt} = 0 \rightarrow \frac{d\phi_{out}}{dt} - \frac{d\phi_{in}}{dt} = 0 \tag{1.1}$$

since

$$\omega = \frac{d\phi}{dt} \tag{1.2}$$

it can be concluded that in phase lock

$$\omega_{out} = \omega_{in} \tag{1.3}$$

As such the PLL tries to reproduce the input signal, which is useful in several situations. For example in case of an incoming clock signal which suffers from period jitter.

Though the base frequency is supposed to be constant, due to phase noise this is not the case. A low pass filter (LPF) is usually added to the loop (see Figure 1.1). This filter suppresses the phase noise of the input signal, the control voltage of the VCO will be more constant than without this LPF. Therefore the high frequencies, due to, for example, phase noise, will be suppressed and less available at the output of the PLL. The VCO reproduces the incoming signal and it is assumed that this VCO itself creates less period jitter than is available at the input of the PLL.

Another reason to add such an LPF is that typically the PD output signal contains both a dc-component and high-frequency components. This is partly due to the period jitter but due to the implementation of the PD itself as well. As can be seen in figure 1.2 V_{PD} must contain a dc-component and high-frequency components. Since rapid fluctuations at the input of the VCO cause the VCO to vary as well, those high frequencies produce extra period jitter.

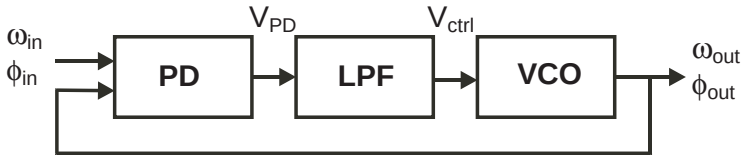


Figure 1.1: Typical PLL

1.1.1 Dynamics of a basic PLL - type I and II

To analyze what a PLL exactly does it is worthwhile to do some s-domain derivations. Since a PLL in general compares input and output phases $\Phi(s)$ will be of particular interest. So let us start with every individual stage of figure 1.1.

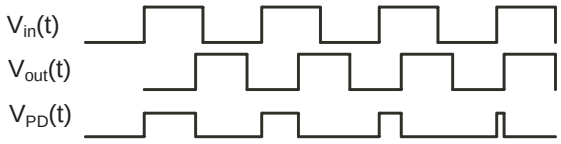


Figure 1.2: Output signal of a simple phase detector

The phase detector compares both $\Phi_{in}(s)$ and $\Phi_{out}(s)$ by subtracting them with the excess phase error $\Phi_e(s)$ as a result. In practice however this is combined with a certain gain K_{PD}.

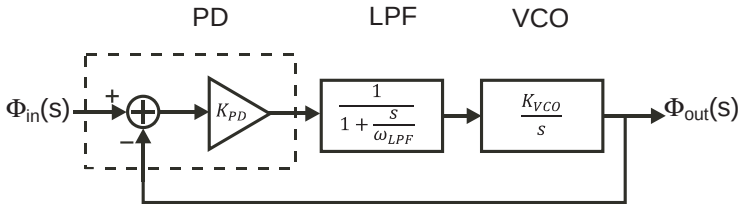


Figure 1.3: General phase detector with gain K_{PD}

The LPF could be a simple RC-network for example, which will result in a *type I* PLL as explained later on. Such an LPF has a -3dB-bandwidth of ω_{LPF}.

For the VCO has as output ω_{out} and the PD expects a phase this output signal should be integrated according to the reverse of Equation 1.2. In s-domain this means

$$\Phi_{out}(s) = \frac{\omega_{out}}{s} \tag{1.4}$$

A general feedback system is depicted in Figure 1.4. So $H(s)$ results in

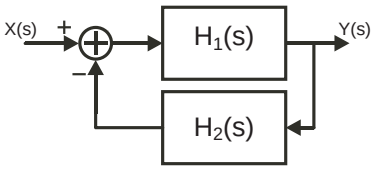


Figure 1.4: General negative feedback system

$$H(s) = \frac{Y(s)}{X(s)} \tag{1.5}$$

$$= \frac{H_1(s)}{1 + H_1(s)H_2(s)} \tag{1.6}$$

The PLL of Figure 1.3 has transfer function

$$H(s) = \frac{K_{PD}K_{VCO}\omega_{LPF}}{s^2 + \omega_{LPF}s + K_{PD}K_{VCO}\omega_{LPF}} \tag{1.7}$$

or more general

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{1.8}$$

with

$$\omega_n = \sqrt{K_{PD}K_{VCO}\omega_{LPF}} \tag{1.9}$$

$$\zeta = \frac{1}{2}\sqrt{\frac{\omega_{LPF}}{K_{PD}K_{VCO}}} \tag{1.10}$$

Another typically used building block next to the LPF is a charge pump. Every cycle that the output phase differs from the input phase the amount of charge (and thus V_{ctrl}) will be adjusted. The phase/frequency detector (PFD)

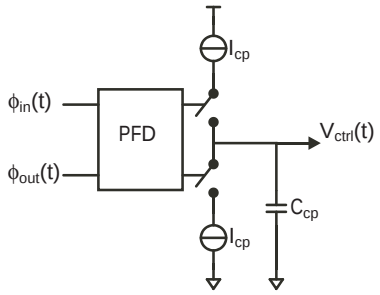


Figure 1.5: General charge-pump for a PLL

compares again ϕ_{in} and ϕ_{out} and steers the switches of the charge pump in order to change the amount of charge in C_{CP}. The transfer function of this PFD is equal to

$$\frac{V_{PFD}(s)}{\Phi_e(s)} = \frac{I_{CP}}{2\pi C_{CP}}\frac{1}{s} = K_{PFD}\frac{1}{s} \tag{1.11}$$

This means that this type of low pass filtering has a pole at the origin of the s-plane. As the VCO already has a pole at the origin too this will be a PLL of *type II*, for there are two poles in the origin of the open loop transfer function. A *type I* PLL has only one pole in the origin. The overall transfer function results in

$$H(s) = \frac{K_{PFD}K_{VCO}}{s^2 + K_{PFD}K_{VCO}} \tag{1.12}$$

It can easily be seen that this system will start to oscillate as there are two poles at the imaginary axis. In general this can be solved by placing an extra resistance in series with the capacitor to create a zero. The root locus of the double pole at the origin will bend towards the left-half plane.

$$H(s) = \frac{K_{PFD}K_{VCO}(RC_{CP}s + 1)}{s^2 + K_{PFD}K_{VCO}C_{CP}Rs + K_{PFD}K_{VCO}} \tag{1.13}$$

ω_n and ζ would then be

$$\omega_n = \sqrt{K_{PFD}K_{VCO}} \tag{1.14}$$

$$\zeta = \frac{R}{2}\sqrt{K_{PFD}K_{VCO}C_{CP}^2} \tag{1.15}$$

1.2 Problem definition

Second-order systems can be described generally as follows

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{1.16}$$

The poles of such a system can be found at

$$s_{1,2} = \frac{-2\zeta\omega_n \pm \sqrt{4\zeta^2\omega_n^2 - 4\omega_n^2}}{2} \tag{1.17}$$

$$= -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1} \tag{1.18}$$

and if $\zeta < 1$ equation 1.18 turns into

$$s_{1,2} = -\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2} \tag{1.19}$$

As can easily be seen the real term $-\zeta\omega_n$ defines the *absolute damping* since in time-domain the poles convert to

$$y(t) = e^{-\zeta\omega_n t}e^{\pm j\omega_n\sqrt{1-\zeta^2}t} \tag{1.20}$$

ζ is called the *relative damping* since it defines the shape of the impulse response.

So in general, the term $-\zeta\omega_n$ contributes to the *settling time* of a second-order PLL. Recall though that ϕ_{in} and ϕ_{out} were assumed to be comparable,but if $\omega_{out}/\omega_{in} \neq 1$ this is not the case. So first the PLL has to 'walk through' all frequencies until $\omega_{out}/\omega_{in} \approx 1$ and then it starts to lock the phase. This can be compared by two *flywheels*.

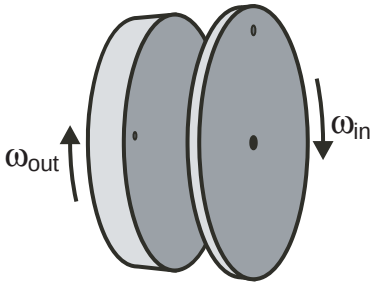


Figure 1.6: A PLL as a pair of flywheels

The incoming signal can be compared with a thin flywheel with a small hole in it to determine the phase. The output frequency of the PLL can be compared with a much heavier flywheel with a little hole as well. As the input signal is available ω_{out} is still equal to zero, so the PLL detects that both holes (phases) are not equally positioned. So ω_{out} has to be tuned up, but because of its mass this will take a lot of time.

Let one derive the amount of time necessary. Phase-lock implies $\omega_{out} = \omega_{in}$. But the reverse is not true: if $\omega_{out} = \omega_{in}$, $\phi_{out} \neq \phi_{in}$ can be true as well. This can easily be seen with the example of the flywheels. If both turn around at the same speed, the little holes can still be at different places.

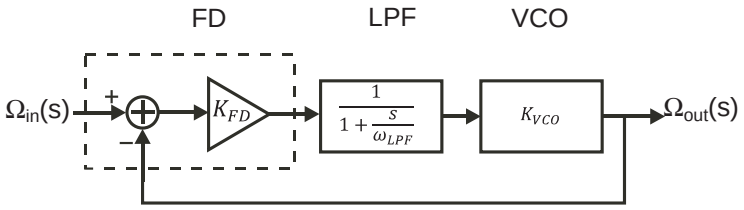


Figure 1.7: Frequency-locked loop (FLL)

So let one consider the frequency response instead of the phase response. Assume that there is only a frequency detector available with parameter K_{FD} comparing ω_{out} and ω_{in}.

$$H(s) = \frac{\Omega_{out}}{\Omega_{in}}(s) \tag{1.21}$$

$$= \frac{K\omega_{LPF}}{s + \omega_{LPF}(1+K)} \tag{1.22}$$

with $K = K_{FD}K_{VCO}$. Translating this to time-domain, the time-dependent output results to contain the function

$$f(t) = e^{-\omega_{LPF}(1+K)t} \tag{1.23}$$

and in general has a plot like[1]

Figure 1.8: Impression of Frequency-Locked Loop in time domain

One could decrease the flywheel's mass by increasing the ω_{LPF} or the gain K. This surely will shorten the settle time, but then the PLL will be more susceptible to noise at the input as the loop bandwidth increases. This is not desirable as the input signal is assumed to suffer from phase noise. The loop bandwidth has to remain as low as possible in order to suppress the phase noise at the input.

This can be compared with the analogy of the flywheel again. As one touches the light weighted flywheel with a finger for a short moment of time, its frequency

[1]Note that in a PLL only once per period information about the current excess phase is obtained. As such equations 1.22 and 1.23 only give an impression of the variables that play a role in the locking process.

would be adjusted and then return to its original ω_{in}. Because of its heavy mass the other flywheel cannot follow the small change in velocity that rapidly, so ω_{out} will remain relatively constant. That is why such a PLL is also called a *clock cleaner*: it literally suppresses the period jitter of the input signal. Therefore the loop bandwidth has to remain as low as possible.

An often seen solution, to shorten this settling time, is an adjustable loop bandwidth [5]. The PLL then rapidly runs through all frequencies with a high loop bandwidth till phase lock and then lowers this loop bandwidth to suppress the phase noise. However, the PLL still has to run through all frequencies up to ω_{in}.

1.3 Proposed solution

A possible solution is to estimate ω_{in} within one period T_{in}, preset the VCO and then start fine locking. Talking in terms of flywheels, this would result in a flywheel with ω_{out} which starts from $\omega_{out} = \omega_{rough}$ instead of $\omega_{out} = 0$. Since the incoming clock signal, including period jitter, gives an good idea of ω_{in} (see Figure 1.9). This could result in a much faster phase-lock.

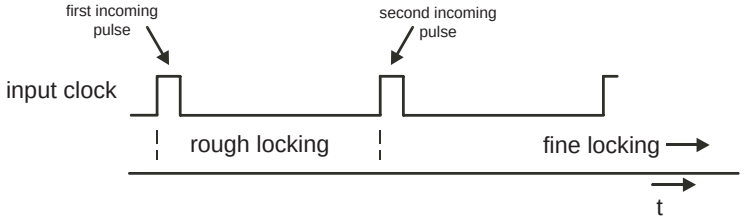


Figure 1.9: Proposal solution

Recall that to start locking the phase properly with $-\zeta\omega_n$ as an indicator for the settling time, ω_{in} and ω_{out} have to be near to each other. By estimating ω_{in} and set ω_{out} to $\omega_{out} = \omega_{rough}$ the system will be faster.

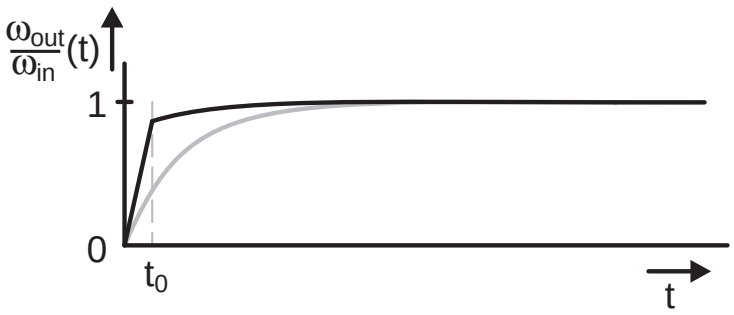


Figure 1.10: Proposal solution

The idea is to estimate the time between the first incoming clock pulse and the second, which gives $\widetilde{T}_{in}$. This does not give the exact value of T_{in} since the incoming signal is assumed to be jittered. The advantage though, as can

be seen in figure 1.10, is that the PLL is within t_0 seconds already almost at $\omega_{out}/\omega_{in} = 1$ which is faster than the former option of the figure.

1.4 Objective

Aiming at functionality of such a clock cleaner it is sensible not to put effort in high frequency behavior while not knowing whether a novel idea will be feasible. Therefore to aim at relatively low frequencies will be sufficient. In modern processes at low frequencies parasitics will be negligible which gives ultimate possibility to aim at functionality only. After realizing a certain design, one can go for higher frequencies.

In CMOS065 a frequency range of 1MHz to 10MHz could be implemented easily without worrying too much about parasitics. These frequencies would already be interesting for low speed communications such as digital audio.

Since there will be phase noise present at the incoming signal an infinitely precise estimation of the frequency will be a waste of energy and time. Generally an oscillator does not produce more phase noise than a few parts per million. Due to bad circuit design, modulation of the clock, etcetera, the phase noise can be much higher. A maximum of one percent period jitter is already quite a lot, so five percent should cover most signals. So a good point to aim at is relative period jitter of 0% to 5%.

In summary:

1. frequency range : 1-10MHz
2. relative period jitter : 0-5%

Chapter 2

Proposed solution exploration

The basic idea of this PLL with frequency estimator is to make a rough estimation of the incoming base frequency. A standard PLL can be compared with a flywheel which starts from 0 $rads^{-1}$ slowly tuning up to the same frequency and phase as the incoming signal. The principle of a PLL with frequency estimator is to give the flywheel an initial frequency ω_{rough}.

In order to do this the system should walk through different phases. First a start-up needs to be done in order to set the PLL ready to wait for an incoming signal. As the system will be waiting for an incoming signal which may arrive at an arbitrary moment in time, the power consumption must be kept as low as possible. This phase is called Initialization Phase (I-Phase).

As an signal comes in the system should do perform a rough estimation within one clock period of the incoming signal, this phase is called the Rough Locking Phase (RL-Phase).

After this estimation the systems performs a well known PLL operation minimizing the excess phase ϕ_e. This phase is called the Fine Locking Phase (FL-Phase).

The system should follow a certain state flow in order to work properly. Generally three phases can be distinguished.

1. Initialization Phase
2. Rough Locking Phase
3. Fine Locking Phase

2.1 Estimation principle

Basically one needs a section which handles the incoming signal (control logic (CL)) and an oscillator (OSC) to take care of the output signal. So the CL stage has to do some estimation of the frequency and preset the OSC before starting the conventional phase lock method. In order to do such an estimation the idea is to charge a capacitor from the first incoming clock pulse and stop as the second pulse comes in. The voltage across the capacitor represents the

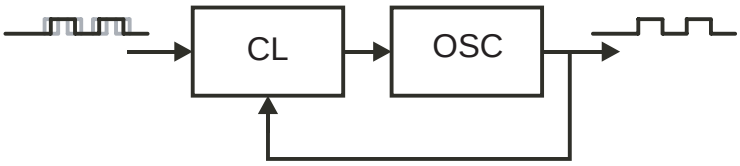


Figure 2.1: System Level PLL

clock's period and as such its frequency. This is the main principle, but one can

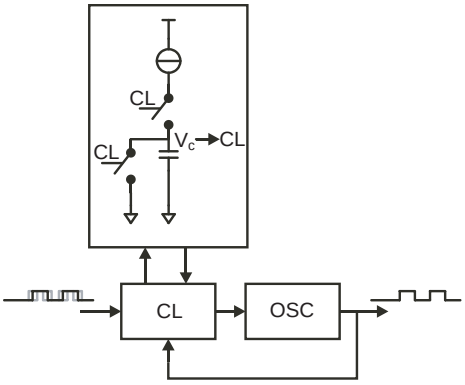


Figure 2.2: PLL with frequency estimator

also realize a Time-to-Digital Converter (TDC) in order to determine ω_r. This is more or less the same, but the capacitor is split up into several parts and are integrated with transistors by use of inverters.

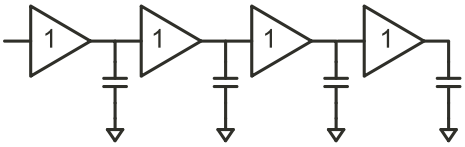


Figure 2.3: Time-to-Digital Converter

The advantage of this TDC is that the estimation can be digitally read out. To make the estimation more accurate (without using more inverters) the array can be followed by a counter. Once this counter has reached its maximum value the process starts over again counting the number of times that the signal flew through the TDC. This is repeated until the next clock pulse comes in. In this way one reuses hardware and can still be accurate.

Implementing a frequency estimator which is separated from the control logic and the oscillator has as an advantage that in principle every oscillator can be connected. Only the control logic 'sees' the estimator. As such different types of frequency ranges can be realized using this principle.

However, the drawback is that the estimator needs to be calibrated to the oscillator. Though it's an estimation this could lengthen the locking time, depending on the implementation.

Another possibility is to integrate the frequency estimator with the oscillator. This could then look like the system in Figure 2.4. As can be seen in the figure ϕ_{out} is retrieved by sampling the voltage over the capacitor. The sampling moments will be the rising edges (for instance) of the incoming signal.

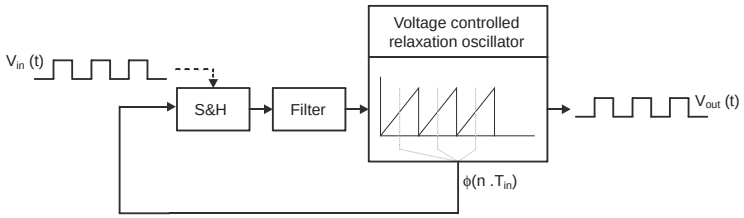


Figure 2.4: PLL with a relaxation oscillator

As already discussed the frequency range to be aimed at consists of relatively low frequencies in the range from 1 MHz to 10 MHz. This could be done by the switched capacitor oscillator [2] which is a relaxation oscillator and as such contains a capacitor which also could be used as a frequency estimator.

In general relaxation oscillators produce a lot of phase noise and are far from ideal. This seems not a good idea since the output signal could contain more phase noise than the input. The switched capacitor relaxation oscillator though has a very good FoM compared to other relaxation oscillators [1]. This enables the possibility to actually clean incoming clock signals at those frequencies.

2.1.1 Switched capacitor relaxation oscillator

The oscillator makes use of the principle of Underhill [3] which says that the reference levels of a sawtooth like curve may by noisy as long as the output trigger circuit lays in between. As can be seen in see figure 2.5 the up going parts of the curve cross the dashed line with equal interval T.

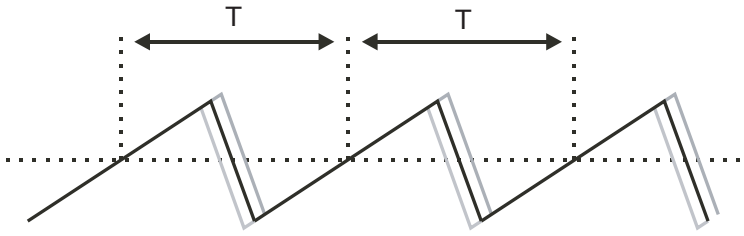


Figure 2.5: Underhill principle

Based on this idea the switched capacitor oscillator has a very good FoM [2]. As short introduction to its operation follows now, but this is more extensively discussed in [2] and in Chapter 3. The main capacitor of this oscillator is capacitor C_1 which is charged by current source I_1. As it reaches a maximum charge level, sensed by the comparator, capacitor C_2 will be reversed in order to discharge C_1 due to charge redistribution. Current source I_2 provides the charge necessary.

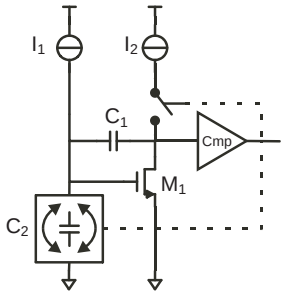


Figure 2.6: Switched capacitor relaxation oscillator

The actual clock signal generated is not the control signal from the comparator drawn in figure 3.1 which decides whether C_1 should be discharged or not. No, the actual output signal is an extra comparator representing the dashed line in Figure 2.5. As can be seen in the figure is that the timing of the discharge does not matter for the period length T.

2.1.2 Relaxation oscillator with integrated frequency estimator

With a relaxation oscillator it would be possible to realize a voltage curve across C_1 like in Figure 2.7. The idea is to charge a capacitor (C_1 in case of the

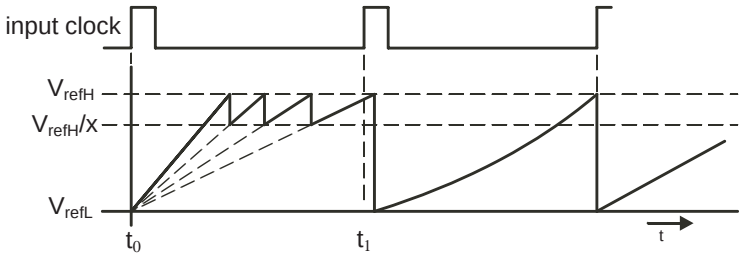


Figure 2.7: Integration of frequency estimator into relaxation oscillator

switched capacitor oscillator) as fast as possible through a charging current I (I_1 in case of the switched capacitor oscillator) as the first clock pulse arrives. Soon enough voltage across this capacitor will reach its upper limit, V_{refH}, and can be concluded that the charging current I was too large. If one then divides the current by a factor x and discharges C from V_{refH} till V_{refH}/x, the new voltage curve over C_1 would be as it has never been different and still points to the origin at t_0 in figure 2.7. This process is repeated until the second clock pulse arrives. Note that this would inherently avoid the problem of calibration between estimator and oscillator.

As the second clock pulse arrives the system could maintain the current value[1] of I_1/x^n in order to measure the value between V_C and V_{refH} which represents the error made by the estimation. So, the estimation continues until ω_{rough} is slightly lower than ω_{in}. And in the second clock period adds the PLL

[1] n is the number of iterations/devisions done

a current I_e to I/x^n in order to correct for the error made during the rough estimation.

As can be seen in Figure 2.7 the voltage over the capacitor has not reached V_{refH} yet as the second pulse arrives. The voltage difference between V_C and V_{refH} gives information about the error of the estimation. Utilizing this difference introduces the so called *Medium Locking Phase* (ML-Phase).

As described in the former chapter the incoming signal is assumed to suffer from a 5% period jitter. However a lot of signals do not suffer from more period jitter than 1%. So the idea is to take 5% period jitter into account in the CL-Phase and 1% in the ML-Phase. This would make it possible to make a more precise estimation.

In terms of the variables of the switched capacitor oscillator the system estimates the best current value for I_1, by charging and discharging capacitor C_1. The ratio ω_{out}/ω_{in} implicitly starts being greater than 1 and within one clock period goes to ω_r/ω_{in} which is slightly lower than 1.

This can be seen in Figure 2.8. While a typical PLL would slowly run through al frequencies, this PLL will do an estimation and starts fine locking from t_0.

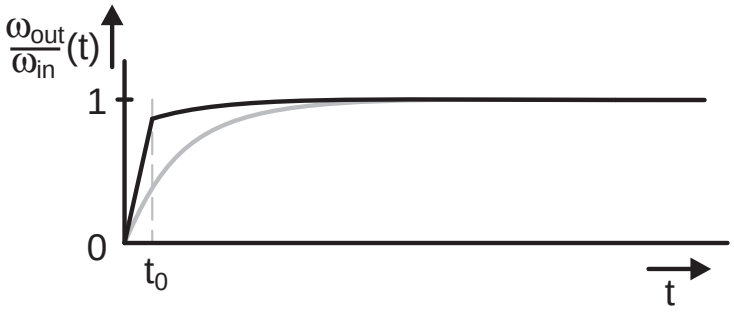


Figure 2.8: Proposal solution

The switched capacitor oscillator will be used as it has the advantage that the phase of the output frequency can be directly synchronized with the phase of the input frequency. See Figure 2.7. As the first incoming clock pulse arrives the relaxation oscillator will start charging the capacitor, hence its output signal's period. This means inherently that the input and output phases are synchronized as the first pulse arrives.

As the RL-Phase finishes, with the arrival of the second incoming pulse, both input en output phase are almost equal. See Figure 2.7. Only a small phase/frequency difference is left over for the FL-Phase. This will mean that even the settling time itself will be shortened as the absolute damping time $(-\zeta\omega_n)$ has already been partly passed. Due to a synchronized start of the oscillator's period the ratio ϕ_{out}/ϕ_{in} always starts close to 1. This is represented by the cross in Figure 2.9.

If the frequency estimator would be externally implemented, such that estimator and VCO are two separate building blocks, this is not trivial. Possibly ϕ_{out}/ϕ_{in} is arbitrarily and an extra circuit could be necessary for synchronization.

As the switched capacitor relaxation oscillator has a very good FoM for a

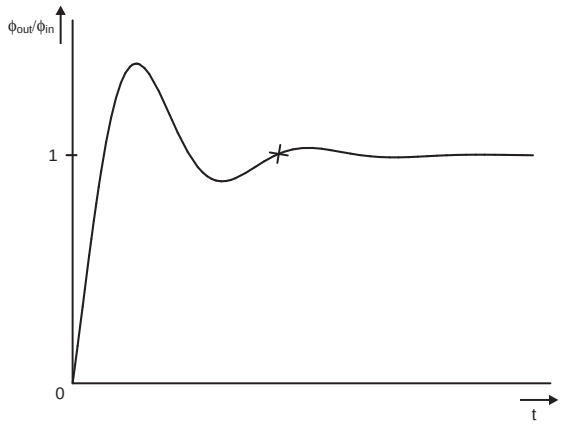


Figure 2.9: ϕ_{out}/ϕ_{in} starts from the cross due to integration of the frequency estimator

relaxation oscillator and the fact that a relaxation oscillator will estimate both the frequency and the phase the switched capacitor relaxation oscillator will be used for this project.

2.2 Summary

Three phases can be distinguished: initialization, rough locking and fine locking. In case of the usage of a relaxation oscillator with integrated frequency estimator an extra phase can be placed in between the rough and fine locking phase, this will be the medium locking phase.

For the rough locking phase 5% period jitter will be used. If a ML-Phase is implemented this percentage could be lowered to 1%.

The switched capacitor oscillator is a good option to start with.

Chapter 3

Oscillator exploration

In order to integrate the estimator with oscillator it is necessary to do some exploration on the oscillator. First some functionality is described, then the FoM and at last the linearity will be investigated. The oscillator appears to be more linear than expected as it consists of non-linear components.

3.1 Operation

A typical relaxation oscillator contains a capacitor which is charged and discharged alternately. In this oscillator that is capacitor C_1 of Figure 3.1[1]. Assume

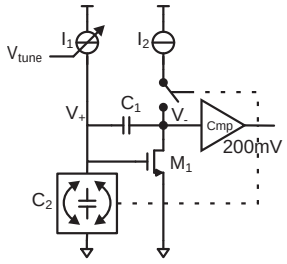


Figure 3.1: Switched capacitor relaxation oscillator

both capacitors to be empty[2], V_+ is set to 0V and as such M_1 is turned off. Assume the switch to be open. At this moment all current from I_1 must flow into C_2 as node V_- has a (apart from C_1) high impedance.

Due to the raising amount of charge in C_2 V_+ will raise. As V_+ reaches V_{th} of M_1, M_1 will start to raise its output current forcing all current of I_1 to flow through C_1 instead of C_2. In this manner there is a fixed charge packet in C_2 and thus from now on a constant[3] $V_+ \approx V_{th,M1}$[4] (see Figure 3.2).

Since C_1 is being charged its voltage must increase which means that V_- drops for the positive side of the capacitor is connected to the relative constant

[1] Current source I_1 consists of a PMOS transistor which is degenerated with a resistor. The voltage between gate and V_{dd} is called V_{tune}

[2] In practice C_1 is pre-charged to 200mV

[3] The approximate-sign will be explained in Chapter 4

[4] The approximate-sign will be explained in Chapter 4.

V_+. As V_- drops below the reference voltage of the comparator (200mV) C_2 is switched reversely causing V_+ to drop to approximately $-V_{th,M1}$. M_1 will then turn off. A fixed charge packet is supposed to be subtracted from C_1, but as M_1 is turned off this means that current source I_1 will charge C_2 while the amount of charge in C_1 remains affected. Current source I_2 is now turned on in order to provide actual discharging of C_1. The total amount of charge from I_2 can be a little less than necessary compared to the charge packet required by C_2. The remaining necessary amount of charge comes from I_1 with RC-time $C_2/g_{m,M_1}$.

The comparator is allowed to be noisy due to the Underhill principle. Therefore the reference voltage (200mV in this case) may be a simple voltage divider made from resistors. A second comparator is necessary to create the output signal. This comparator is directly placed over capacitor C_1. In this way the noise at the nodes V_+ and V_- take less effect.

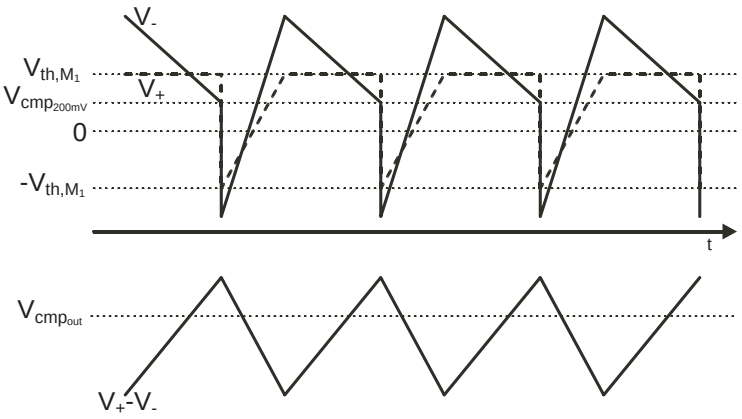


Figure 3.2: Oscillator signals

3.2 FoM

Calculating the FoM only the core energy is of particular interest, which is the energy necessary to perform the actual oscillation. This is the ring of components which keep the oscillator oscillating. In case of this oscillator those are the components necessary to charge and discharge C_1 and the energy consumed by the comparator (as well as its reference voltage) which decides whether the circuit needs to charge or discharge.

The voltages across several components have the following names. The voltage across current source I_1 is called ΔV_1. The voltages across C_1 and C_2 have the names ΔV_3 and ΔV_2 as shown in figure 3.3. Note that the voltages are regarded as *allowed voltage swings*[5], which means that due to $V_{th,M1}$ and the switching character of C_2 ΔV_2 will be equal to $\Delta V_2 \approx 2 \cdot V_{th,M1}$.

Though the Underhill principle takes care of the phase noise due to timing issues of the comparator (seen in Figure 3.1 and 2.5), the charge packet may still vary and affect the phase noise.

[5]This is in the case of Equation 3.1, in the rest of the report they are referred as being *voltage swings*

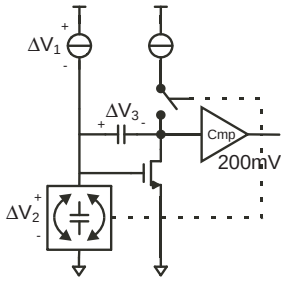


Figure 3.3: Nomenclature

The figure of merit can be calculated as follows [1]

$$\begin{aligned} FoM &= \pounds(f_m)(\frac{f_m}{f_{osc}})^2 P_{core} \cdot 10^3 \\ &= 2kT \cdot \frac{P_{core}}{I_1 \Delta V_{eff}} \cdot 10^3 \end{aligned} \tag{3.1}$$

with

$$\Delta V_{eff} = \frac{\Delta V_1 \cdot \Delta V_2}{\Delta V_1 + \Delta V_2} \tag{3.2}$$

Further details about the switched capacitor oscillator can be found in [2].

Chapter 4

Linearity

Remarkably the oscillator appeared to be more linear than expected. Since the oscillator consists of two non-linear transistors (of which one is degenerated to linearize its current), one would expect the oscillator to show some non-linear behavior. Interestingly though this is not the case as can be seen in Figure 4.1. The figure shows a relative constant K_{VCO}. Assuming a constant voltage swing

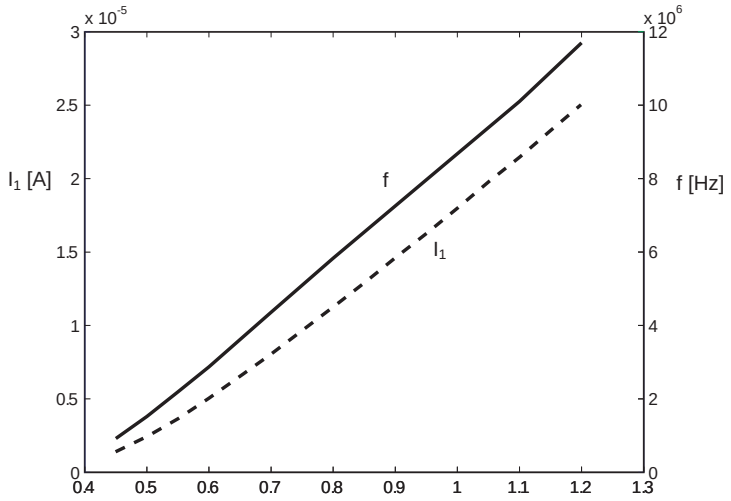


Figure 4.1: The frequency f (solid line) appears to be more linear than I_1 (dashed line) as function of V_{tune}

ΔV_2 due to a semi-constant V_{GS,M_1}[1] over C_2 one would expect a non-linear relation between V_{tune} and the output frequency f.

This is explained as follows. A certain amount of charge will be taken from C_1, when C_2 is reversed. This amount of charge is called Q_d and is equal to $Q_d = C_2 \Delta V_2$. The time necessary to charge C_1 again to its former level is equal to the time necessary to charge C_2 again to its former level, i.e. to provide Q_d

[1]Though M_1 is turned off and on again as C_2 is being switched, V_{GS,M_1} settles to a constant value after switching. That's why V_{GS,M_1} is called semi-constant

by I_1. This amount of time is equal to $t_{charge} = Q_d/I_1$ in which $Q_d = C_2\Delta V_2$. As such the frequency would be equal to

$$f = \frac{I_1}{C_2\Delta V_2} \tag{4.1}$$

This means that the frequency would be as (non)linear as I_1 is, assuming C_2 and ΔV_2 to be constant. The frequency curve would then be proportional to the curve of I_1. However, as can be seen in Figure 4.1, this is not the case.

Current source I_1 is degenerated and as such more linear in the upper region of V_{tune} than for the values nearby V_{TH,I_1}. However for the lower values of V_{tune} the frequency appears to be more linear than I_1.

In the following sections several models will be applied in order to investigate the origin of this linearity. This could be of use in order to integrate a frequency estimator with the oscillator.

Somehow either C_2 or ΔV_2 is not constant. It can be easily seen that $\Delta V_2 \neq 2{\cdot}V_{th,M_1}$ but $\Delta V_2 = 2{\cdot}V_{GS,M_1}$ which is dependent of I_1. The possibility whether this would cause the linear frequency dependance will be investigated in the following two sections 'Square-law equations' and 'Subthreshold equations'.

Section 'Curve-fitted model' deals with C_2, which gives the answer. The other two sections are written to show the reader that the variation of ΔV_2 due to I_1 does hardly contribute to the linearity of the frequency.

4.1 Square-law equations

Though working in CMOS065 both transistors, M_1 and the transistor for I_1, have been designed $1\mu m$ long, which enables the research on the linearity to be quite easy starting with first order MOS models. As two nonlinear devices could cancel their nonlinearity this option will be modeled first. Though the current source I_1 is degenerated, this is to show that indeed both transistors cancel each other's nonlinearity. But only for values of V_{tune} that are higher than V_{dd}. Afterwards I_1 indeed will be modeled being degenerated.

4.1.1 Non-degenerated

As already mentioned the frequency, determined by I_1, has its own charge packet, determined by V_{GS,M_1}. According to Equation 4.2 the relation contains a square root.

$$I_{d,M_1} = \frac{1}{2}\beta(V_{GS} - V_{th})^2 \rightarrow V_{GS,M_1} = \sqrt{\frac{2I_d}{\beta}} + V_{th} \tag{4.2}$$

so assuming $I_{d,M_1} = I_1$, the following equation for the frequency can be derived

$$f = \frac{I_1}{\frac{2\sqrt{2}C_2}{\sqrt{\beta}}\sqrt{I_1} + 2C_2V_{th}} \tag{4.3}$$

Now let one assume that the non-degenerated current source I_1 is a PMOS dimensioned such that $\beta_{I_1} = \beta_{M_1} = \beta$ and $V_{th,I_1} = V_{th,M_1} = V_{th}$ with current relation

$$I_1 = \frac{1}{2}\beta(V_{tune} - V_{th})^2 \tag{4.4}$$

then equation 4.3 transforms into

$$f = \beta \frac{V_{tune}^2 - 2V_{tune}V_{th} + V_{th}^2}{4C_2 V_{tune}} \tag{4.5}$$

If $V_{tune} \gg V_{th}$ then

$$\begin{aligned} f &\approx \beta \frac{V_{tune}(V_{tune} - 2V_{th})}{4C_2 V_{tune}} \\ &= \frac{\beta}{4C_2}(V_{tune} - 2V_{th}) \end{aligned} \tag{4.6}$$

Acquiring values forβ_{I_1}, $= \beta_{M_1}$, V_{th,I_1} and V_{th,M_1} from ProMOST and use them in a MAPLE model, in which current source I_1 and transistor M_1 are both explicitly modeled with their individual values like in the oscillator, one acquires the result seen in Figure 4.2.

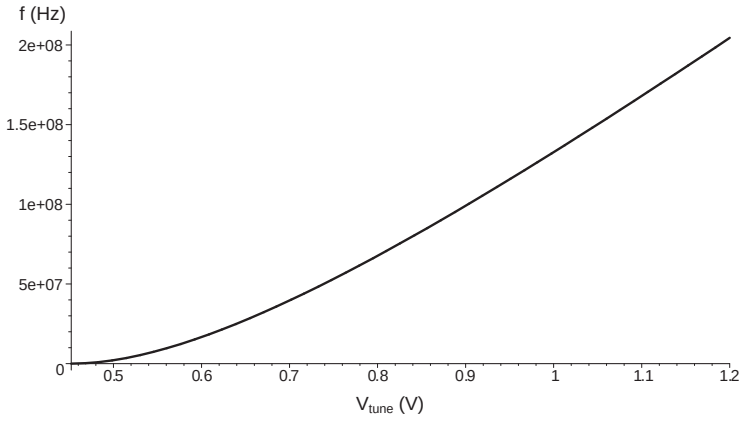


Figure 4.2: Frequency calculation with non-degenerated current source

Requiring V_{th}^2 ten times smaller than the rest of the numerator of Equation 4.5 (and so probably negligible), the frequency would be proportional to V_{tune} according to Equation 4.6. The value of V_{tune} must meet the following constraint $V_{tune} \geq 1,73V$. According to Figure 4.2 the condition $V_{tune} \gg V_{th}$ seems to get valid already from $V_{tune} \approx 1V$. With a maximum voltage of $Vdd = 1,2V$ this means that, though both transistors cancel each other with respect to non-linearity, this cannot be the case with the actual oscillator.

4.1.2 Degenerated

Observing a degenerated current source, I_1 is assumed to be linear from $V_{tune} \approx 0,6V$. So assuming $I_1 \approx G_m V_{tune}$, with $G_m \approx 32,5mS$ (according to simulation results[2]), equation 4.3 transforms into

$$f = \frac{G_m V_{tune}}{\frac{2\sqrt{2}C_2}{\sqrt{\beta}}\sqrt{G_m V_{tune}} + 2C_2 V_{th}} \tag{4.7}$$

$$= \delta_1 \frac{V_{tune}}{\sqrt{V_{tune}} + \delta_2} \tag{4.8}$$

[2]Double gate PMOS $W/L = 29/1$ and $R_{deg} = 24k\Omega$

with

$$\delta_1 = \frac{\sqrt{2G_m\beta}}{4C_2} \tag{4.9}$$

$$\delta_2 = V_{th}\sqrt{\frac{\beta}{2G_m}} \tag{4.10}$$

where δ_2 turns out to be approximately $3\sqrt{V}$ (according to simulation results), which means that within the range $V_{tune} = 0,6\ldots1,2V$ the frequency indeed shows an almost proportional relation to ${V_{tune}}^3$, see Figure 4.3. So the rela-

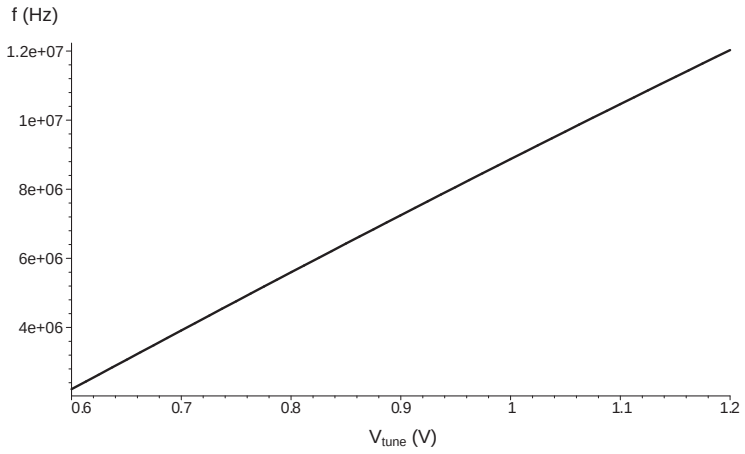


Figure 4.3: Frequency calculation with degenerated current source

tion between frequency and V_{tune} shows an almost linear behavior at the range $V_{tune} = 0,6\ldots1,2V$.

Using Equations 4.1, 4.2 and 4.4 to derive a frequency model with a degenerated current source one acquires the plot in Figure 4.4. The figure shows a

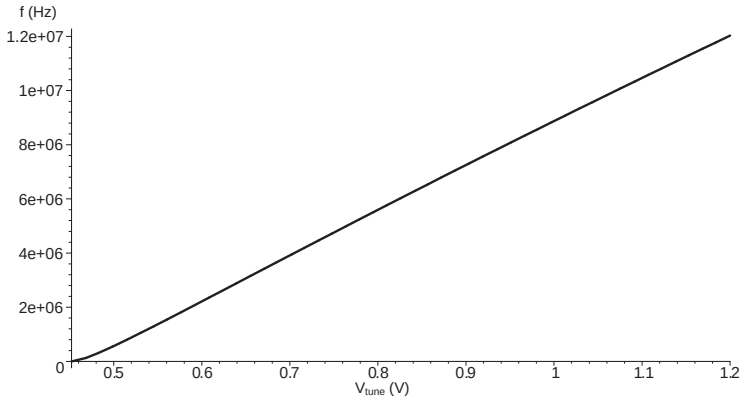


Figure 4.4: Frequency calculation with degenerated current source

[3] $G_m = 35\mu S$, $\beta = 4m\frac{A}{V^2}$ and $V_{th} = 400mV$

linear relation for a wide range of V_{tune}, except for $V_{tune} \approx V_{th,I_1}$. In this region Figures 4.4 and 4.1 show a subtle difference.

Though the square-law models with a degenerated I_1 already explain a linear relation between V_{tune} and the frequency for most values of V_{tune}, there still remains a subtle difference for the very low values of V_{tune}.

4.2 Subthreshold equations

This subtle difference might be explained with the usage of subthreshold equations. The square-law equations do not apply around V_{th} since both transistors operate in weak inversion for the lower values of V_{tune}. This means that the current flowing is mainly due to subthreshold conduction [8].

$$I_d \approx I_0 e^{\frac{V_{GS}-V_{th}}{\zeta V_T}} \tag{4.11}$$

Evaluating equation 4.1 with this equation, the result looks like[4]

$$f \approx \frac{I_0 e^{(V_{GS}-V_{th}/\zeta V_T)}}{2C_2(\zeta V_T log_e(I_d/I_0) + V_{th})} \tag{4.12}$$

The MAPLE results for Equation 4.12 with a degenerated current source[5] I_1 reveal the plot of Figure 4.5. The dotted curve shows f for a degenerated

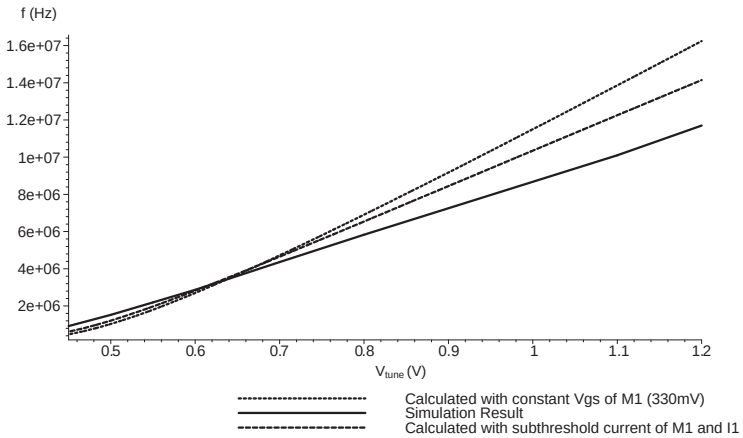


Figure 4.5: Frequency calculation with degenerated current source in subthreshold as well as M_1

current source I_1 with subthreshold relation and a constant ΔV_2. The dashed curve meets the simulation results (solid curve) already better.

Both degenerated current source I_1 and $\Delta V_2 = 2V_{GS,M_1}$ were modeled with subthreshold relations. The reader might notice that both calculated curves are, in the lower region of V_{tune}, still less linear than the simulation results. Moreover that f in the upper part of V_{tune} appears to be too high. So, this (rather simplistic) equation does not reveal convenient results either.

[4]This equation still does not take the degeneration into account

[5]Which is too large to display and does not give much more insight

4.3 Curve-fitted model

Somehow it seems that equation 4.1 needs a little bit more sophistication. What would the frequency plot look like if one would use some curve fitted models for both I_1 and M_1 assuming all of current I_1 flows through M_1.

The curve-fitting was done as follows. Current source I_1 was simulated according to Figure 4.6, assuming V_+ constant and equal to $400mV$. The simula-

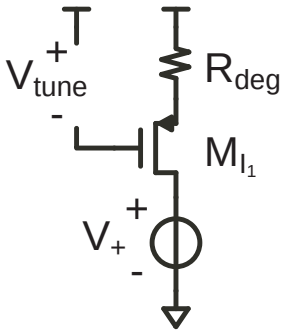


Figure 4.6: Current source I_1 of the switched capacitor oscillator

tion results for the relation between V_{tune} and I_1 were saved and and converted to a mathematical expression with help of the curve fitting function of MAPLE. The same was done for M_1.

It appears that even then the equation does not hold as can bee seen in Figure 4.7. The difference between both frequency plots is plotted in Figure 4.7

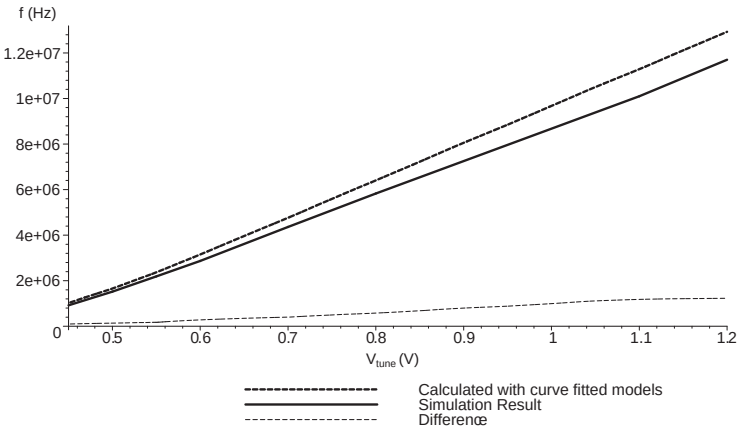


Figure 4.7: Frequency calculation curve fitted models for both I_1 and M_1

as well and requires particular exploration. See figure 4.8. The reader might notice that if V_{tune} increases, the frequency deviates more and more from the simulated results. Since the models used for I_1 and ΔV_2 $(= 2 \cdot V_{GS,M_1})$ were curve fitted, the only variable left in Equation 4.1 is[6] C_2.

[6]The curve-fitted model for I_1 assumes a constant ΔV_2, which is not de case in the oscillator. However, as the PMOS for I_1 is a long device $(1\mu m)$, this variation is expected to be negligible with respect to the frequency

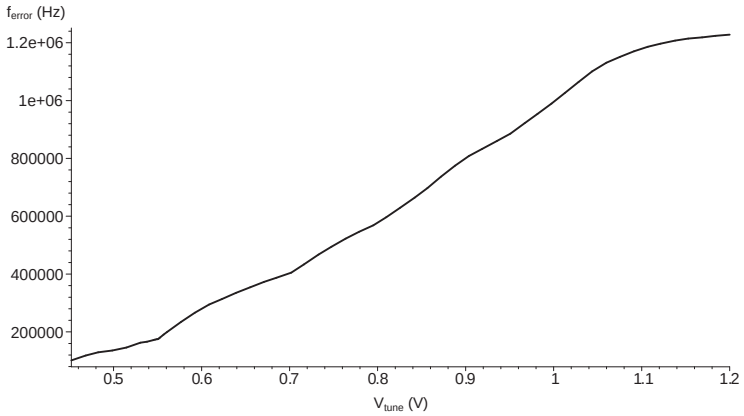


Figure 4.8: Frequency Error

Recall that for the whole range of V_{tune} transistor M_1 operates around the threshold voltage V_{th}.

Figure 4.8 suggests that a C_{GS} and/or C_{GD} could be involved, as over the whole range of V_{tune} transistor M_1 operates in weak inversion C_{GS} and C_{GD} vary. That this could influence the frequency seems plausible since C_{GS,M_1} is in parallel with C_2. As such $\widetilde{C}_2$ becomes larger as V_{tune} increases and the frequency will not increase as much as when C_2 is constant. Note that for the lower values of V_{tune} f is approximately equal to Equation 4.1 as can be seen in Figure 4.7.

Recall that

$$f = \frac{I_1}{C_2 \Delta V_2} \tag{4.13}$$

which means that the value of C_2 determines the frequency as well. Note that C_1 does not appear in this equation, which seems obvious but implies an important conclusion. Though both C_{GS} and C_{GD} of M_1 vary with V_{tune} (and so leaving both C_2 and C_1 varying with V_{tune}) only C_{GS} affects the frequency. This will be according to

$$f = \frac{I_1}{(C_2 + C_{GS}) \Delta V_2} \tag{4.14}$$

Values for C_{GS} obtained by simulation results for a separate transistor equally dimensioned to M_1 reveal the results of Figure 4.9.

C_{GS} values vary from $18fF$ till $215fF$ which is almost 10% of the value of C_2. The gate-source overlap capacitance C_{GSol} was taken into account as well but is constant and with $7,5fF$ negligible. C_{GD,I_1} is in fact parallel with C_2 as well, but is with about $0,1fF$ negligible as well.

So this nonlinearity realizes that the curvy behavior for the lower values of V_{tune} will appear less curvy. As exaggeratedly drawn in Figure 4.10.

So C_{GS} can be up to $215fF$ which is quite large in comparison with the value of C_2 which is $2,5pF$. Increasing C_2 would make the oscillator less sensitive to C_{GS} but more nonlinear as well.

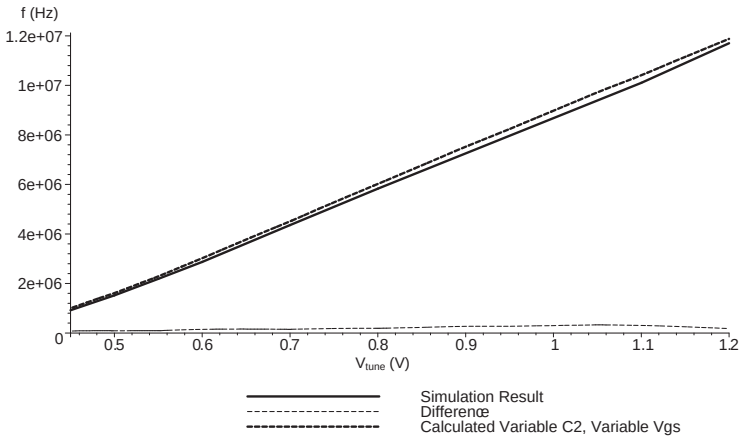


Figure 4.9: Frequency calculation with C_{GS} taken into account

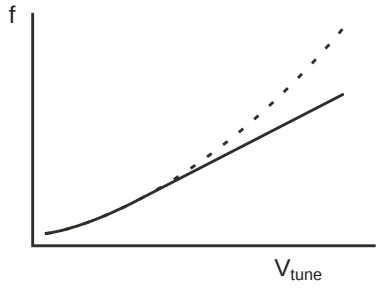


Figure 4.10: More 'linear' relation - an exaggerated example

4.4 Summary

In order to explain the linearity of the oscillator various approaches to model the oscillator's behavior have been explored. The square-law model shows that the $f(V_{tune})$ relation should be linear within the range of $V_{tune} = 0,6 \ldots 1,2V$. But it still shows a subtle difference near the threshold voltage of I_1, which is due to the square-law model which is not accurate enough in this particular region.

Subthreshold relations did not show convenient results either. This is probably due to the rather simplistic relation of Equation 4.11.

Using Equation 4.1 and curve-fitted relations to model I_1 and M_1 revealed that Equation 4.1 needs some sophistication. It appears that not only C_2 determines the frequency, but C_{GS,M_1} as well. This latter capacitance is nonlinear with respect to I_{d,M_1} and thus V_{tune}. This nonlinearity realizes a less curvy relation between f and V_{tune}.

Another fact is that every frequency, determined by I_1 has its own fixed charge packet according to equation 4.2.

Due to the ratio between C_{GS} and C_2 the effect of a variable C_{GS} is significant.

Chapter 5

Design considerations

This chapter handles several considerations to be done in order to design a PLL with frequency estimator given the constraints of Chapter 1. This is done with help of Figure 5.1. The current source of the switched capacitor oscillator should

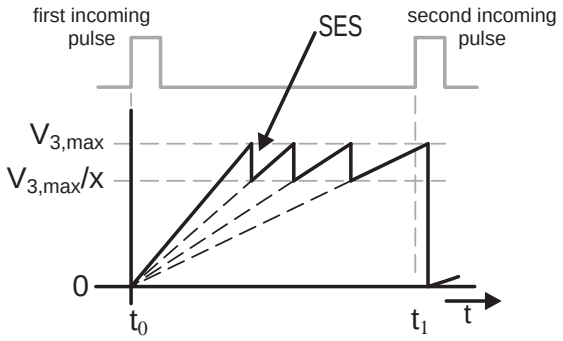


Figure 5.1: Estimation principle

be able to provide various values according to

$$I_{1,CLP} = \frac{I_{max}}{x^n} \tag{5.1}$$

in which n is the iteration number. Afterwards a current I_{tune} should be added in order to perform the FL-Phase.

Another issue is the implementation of the small estimation step (SES) which has to be performed when V_3 reaches $V_{3,max}$ while the second clock pulse has not arrived yet. $V_{3,max}$ should then be divided by the factor x as well. Every time the value of $I_{1,CLP}$ has to be adjusted it will be divided by x, while V_3 will be divided by x as well. As such the slope of V_3 is adjusted such that it seems that it never has been different according to its so called 'new' current. This principle is shown again in Figure 5.2.

Other point of discussion will be the feedback loop which will try to minimize the excess phase ϕ_e.

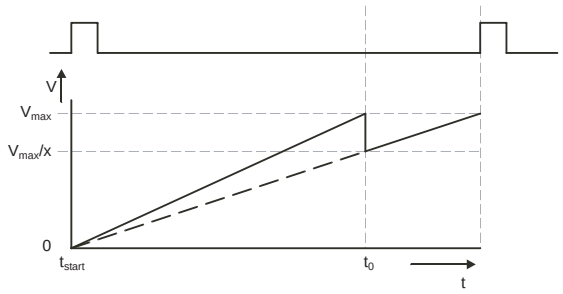


Figure 5.2: Current and voltage divided by x

5.1 Choice of x

One wants to vary the frequency between ω_{min} and ω_{max} in order to do rough locking. This can be done with the variation of current I_1. Every time it has to be adjusted in the RL-Phase, it will be divided by a factor x. This current I_1 will be varied according to Equation 5.1.

If the value of x would be chosen too small, this would lead to an infinitely precise estimation as there will be an infinite number of iterations. If it would be chosen too large, to an estimation which is so rough that it still leaves a lot of work for the FL-Phase. So one wants a value for x which is big enough to perform an efficient estimation.

5.1.1 Precision of estimation

In order to find a value of x recall that the incoming signal suffers from period jitter. This means that if the estimation of the period would be infinitely precise the value for T would still be an estimation as T has a certain spread, see Figure 5.3. This means that $\omega_{in} = 2\pi/(T_{in} + \Delta T_{in})$, where T_{in} stands for the period of the incoming signal and ΔT_{in} is the random variable for the period jitter in *seconds*.

As period jitter is caused by several processes, one can assume the jitter to have a gaussian distribution through the central limit theorem [6]. As such $\Delta T_{in} = \sigma_{in}$

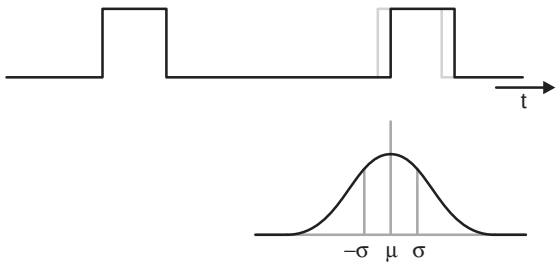


Figure 5.3: Period jitter

So, if one wants to distinguish two frequencies from each other, the problem will be that those two can be very near to each other such that their ΔT_{in}

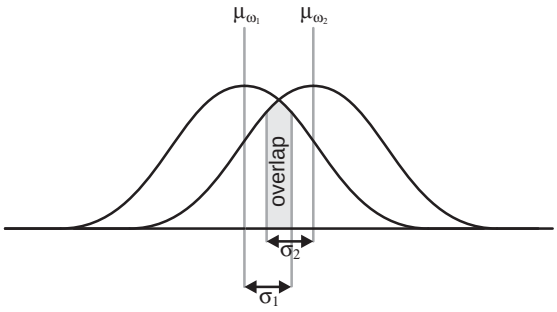


Figure 5.4: Overlap of two near frequencies

will overlap. This can be seen in Figure 5.4. This means that if an incoming pulse arrives one cannot say unambiguously whether it belongs to ω_1 or to ω_2. However, there is a certain chance that an incoming pulse belongs to ω_1 for instance. This chance is equal to 68.3% if the pulse arrives σ seconds from the average value. This chance increases to 99.7% if the pulse arrives within the range of 3σ seconds.

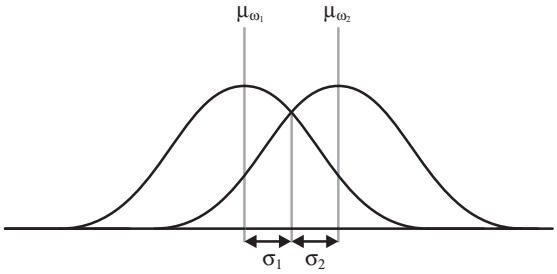


Figure 5.5: Non-overlap of the sigma ranges

If μ_{ω_1} and μ_{ω_2} are chosen such that their spread of 3σ will only touch each other without any overlap (see Figure 5.5), one can always decide whether an incoming pulse belongs to μ_{ω_n} with 99.7% probability[1].

Since every ω_{in} incorporates a certain spread around its value it would not be necessary to do an estimation which is more precise. So there is a certain distance which could define a boundary for the maximum precision of the estimation. This difference is called $|\omega_{in} - \omega_{pj}|$ in which ω_{pj} represents the frequency which ought to be ω_{in} but deviates due to period jitter.

So the minimum difference between ω_{in} and ω_{pj} goes to

$$\begin{aligned} |\omega_{in} - \omega_{pj}| &= 2\pi \cdot |\frac{1}{T_{in}} - \frac{1}{T_{in} - \Delta T_{in}}| \\ &= 2\pi \cdot |\frac{1}{T_{in}} \frac{\Delta T_{in}}{T_{in} - \Delta T_{in}}| \\ &= \frac{p}{1-p} \cdot \omega_{in} \end{aligned} \tag{5.2}$$

[1] In the figure only σ is drawn instead of 3σ

with $p = \Delta T_{in}/T_{in} = \sigma_{in}/T_{in}$, the ratio representing the period jitter.

This equation describes the frequency range in which one could decide that ω_{in} belongs to a certain estimated frequency ω_{pj} with 68.3% probability. In order to gain 99.7% probability Equation 5.2 turns into

$$|\omega_{in} - \omega_{pj}| = \frac{3p}{1-3p} \cdot \omega_{in} \tag{5.3}$$

with $3p = 3\sigma/T_{in}$.

5.1.2 Determination of the rough frequencies

The principle to estimate the incoming frequency is to start with an initial frequency ω_0, see Figure 5.6. The system uses a number of predefined rough

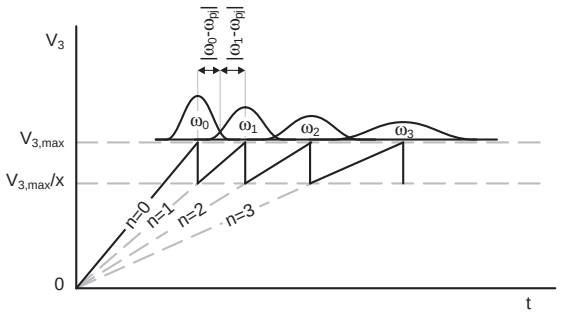


Figure 5.6: Non-overlap of the sigma ranges

frequencies, ω_n, which[2] are determined by

$$\omega_n = 2\pi \cdot \frac{I_{max}}{x^n(C_2 + C_{GS})\Delta V_2} \tag{5.4}$$

$$\sim \frac{1}{x^n} \tag{5.5}$$

Starting with $n = 0$ (the highest frequency in the RL-Phase) the estimator goes downwards from $\omega_{max} = \omega_0$ to $\omega_{min} = \omega_N$ in search of the right frequency. Iterating along the frequency range I_{max} will be divided by x each iteration.

In order to choose the rough frequencies, ω_n, let one assume that the received signal can only be either ω_0 or ω_1. The estimator then only has to distinguish two frequencies: $1/T_0$ ($\sim \omega_0$) and $1/T_1$($\sim \omega_1$). See Figure 5.7.

Suppose that the incoming signal has a period of $1/T_0$, V_3 would then reach $V_{3,max}$ exactly when the second pulse comes in. But if that particular pulse would be slightly later, the system would already assume that the pulse belongs to a pulse that corresponds to T_1, as a division by x has already been performed.

The disadvantage is then that the system started good with trying ω_0, but due to the period jitter it decides to assume that the input signal has a frequency of ω_1. Though this will be solved in the FL-Phase, it takes time.

In order to reduce this error of classifying the incoming pulse wrong and the need to fine tune back from ω_1 to ω_0, the slope of V_3 will be chosen a little bit

[2]Note that in this chapter ω_n means *the frequency belonging to the n^{th} iteration of the RL-Phase*

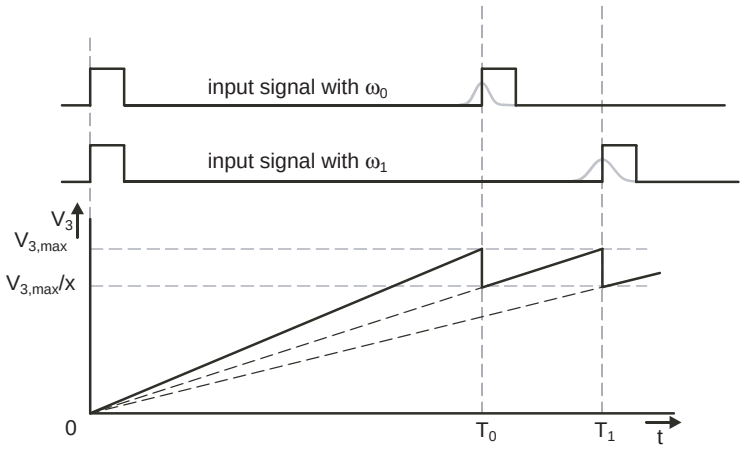


Figure 5.7: Estimation of the frequency

lower such that V_3 reaches $V_{3,max}$ somewhere in the middle of T_0 and T_1. It will be chosen such that it would reach $V_{3,max}$ at $t = T_0 + \Delta T_0$ ($\sim \omega_0 + |\omega_0 - \omega_{pj}|$), see Figure 5.8.

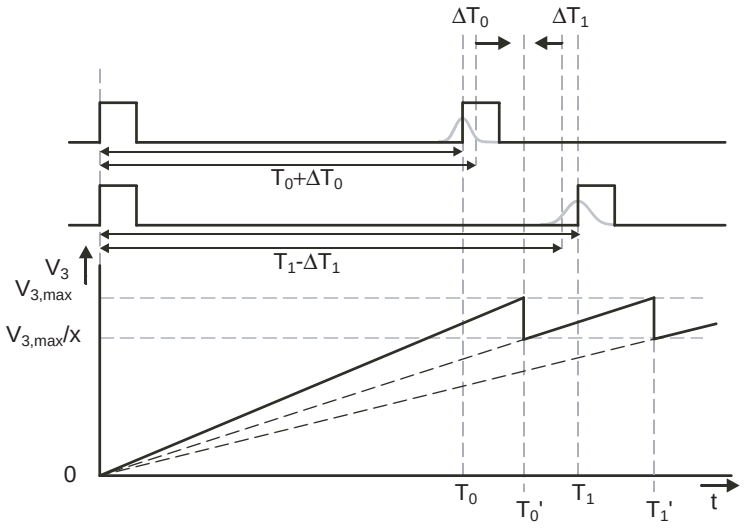


Figure 5.8: The initial frequency of the estimator is chosen between ω_0 and ω_1

5.1.3 Determination of x

Now one knows the spread around a certain ω_{in} and how the rough frequencies ω_n will be chosen, one can calculate the value of x and the maximum number of steps N.

The total factor with which I_{max} is divided after n iterations is α which is equal to

$$\alpha = x^n \tag{5.6}$$

with $n \in [0, N]$ and integer. As such ω_{max} will be divided as well and so one can determine the following results

$$\alpha_{max} = \frac{\omega_{max}}{\omega_{min}} = x^N \tag{5.7}$$

$$N = \frac{\log \omega_{max} - \log \omega_{min}}{\log x} \tag{5.8}$$

where N needs to be rounded up to de next integer value, because of the discrete character of the iteration.

The factor x can then calculated as follows. Locate T_0', T_0 and T_1 such that T_0' becomes the border between $T_0 + 3 \cdot \Delta T_0$ and $T_1 - 3 \cdot \Delta T_1$. This is indicated with the two small arrows in the upper right of Figure 5.8. One can derive the following relations $T_1' - T_0' = 2 \cdot \Delta T_1$, $T_n' = x^n \cdot T_0'$ and $\Delta T_n = 3p \cdot T_n$ and calculate x as follows.

$$x = 1 + \frac{T_1}{T_0'} \cdot 2 \cdot 3p \tag{5.9}$$

and with $T_{n-1}' = T_n - 3\Delta T_n$

$$x = \frac{1 + 3p}{1 - 3p} \tag{5.10}$$

The number of steps to be taken

$$N = \frac{\log \omega_{max}' - \log \omega_{min}'}{\log(1 + 3p) - \log(1 - 3p)} \tag{5.11}$$

and with $T_n = T_n' - 3\Delta T_n$ one can determine that

$$\omega_{max,min}' = \omega_{max,min} \cdot \frac{1}{1 + 3p} \tag{5.12}$$

Equation 5.12 says that in the RL-Phase the oscillator has to be able to handle frequencies a bit lower than the frequencies to estimate. So aiming at the range from 1 to 10MHz with a relative period jitter of $p = 0.05$ this leaves an estimator operating from 0.87MHz up to 8.7MHz.

In fact Equation 5.12 means that in the RL-Phase slightly less current than the actual frequencies is needed. If the system needs to lock to the actual signal, an extra current source is needed which enables the circuit to adjust to the actual amount of current needed.

So with $\Delta V_2 = 800mV$, $C_2 + C_{GS,M_1} \approx 2.7pF$, $f_{max} = 10MHz$ and $p = 0.05$ and combining Equation 4.14 and Equation 5.12 one can calculate the maximum amount of current necessary in the RL-Phase.

$$\begin{aligned} I_{1_{max,CLP}} &= \frac{(C_2 + C_{GS,M_1})\Delta V_2 f_{max}}{1 + 3p} \\ &\approx 18.8\mu A \end{aligned}$$

Calculating the number of divisions to be done using Equation 5.11 one comes to the number of $N = 8$. And $x = 1.35$ using Equation 5.10. The minimum value for I_1 can be calculated by dividing $I_{1_{max,CLP}}$ by α_{max} according to Equation 5.7

$$\begin{aligned} I_{1_{min,CLP}} &= \frac{I_{1_{max,CLP}}}{\alpha_{max}} = \frac{I_{1_{max,CLP}}}{x^N} \\ &\approx 1.67\mu A \end{aligned}$$

5.2 Variable current source I_1

In order to implement a variable current source which has to provide the different values of current I_1, one has different options to implement such a source. Its noise is of importance since it defines the amount of charge in the switched capacitor trough V_{GS,M_1} as can be seen in Figure 5.9. In order to design a

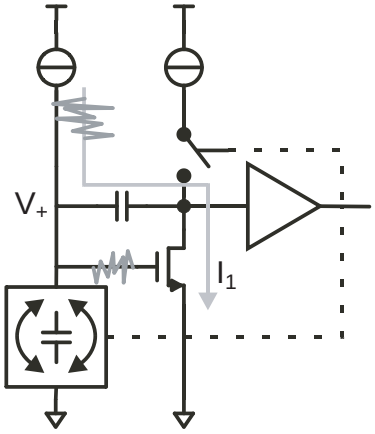


Figure 5.9: Noise in I_1 results in a variable charge packet

current source for the PLL it would be nice to preserve the noise behavior of the oscillator as good as possible.

The current source of the switched capacitor oscillator is depicted in Figure 5.10. The voltage source V_+ represents the voltage over the switched capac-

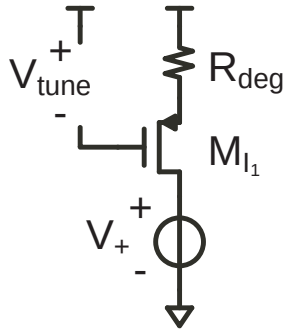


Figure 5.10: Current source I_1 of the switched capacitor oscillator

itor of the oscillator, which is assumed to be constant for a certain value of I_1 when the C_2 is not switched.

The resistor R_{deg} was added in order to dominate the thermal noise of the transistor and to linearize current I_1 and as such the relation between frequency and V_{tune}. This R_{deg} was chosen such that $V_{DS} = V_{DS,SAT}$ for $V_{tune} = V_{tune,max}$. But if the frequency needs to be decreased, current I_1 needs to decrease as well resulting in a lower voltage across R_{deg} leaving the transistor with $V_{DS} > V_{DS,SAT}$.

But even if the voltage across R_{deg} would *somehow* be constant while I_1 varies even then M_{I_1} will not be tuned to $V_{DS,SAT}$. This can be seen in Figure 5.11. If $V_{R_{deg}}$ would be constant, V_{DS} as well, while $V_{DS,SAT}$ varies. The

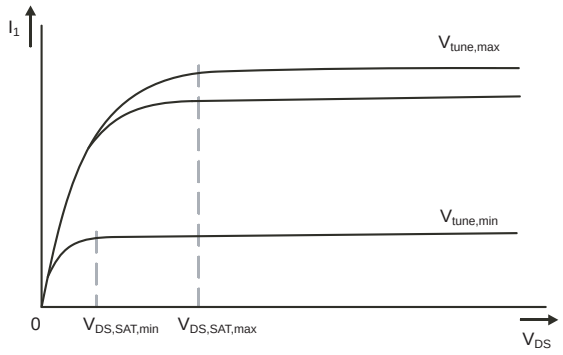


Figure 5.11: $V_{DS} \neq V_{DS,SAT}$ due to change in V_{tune}

transistor would not be tuned to its equivalent noise resistance anymore.

So the option to control the PMOS in an analog way, which seems an easy solution as only one transistor is needed to provide all current needed, incorporates more noise than probably necessary.

Moreover the presence of noise at the gate of this transistor will be fed into the oscillator as well. The noise of the preceding stages will be available at the gate, which is then converted to I_1 through G_m of such a continuous current source.

5.2.1 Semi-discrete current source

It is better to switch the transistor between V_{dd} and ground. Connecting the gate either to ground or V_{dd} reduces the amount of noise present in I_1 as no noisy voltage reference is needed. Multiple transistors can be used to realize the different values of current[3] $I_{1,CLP}$. This gives one the opportunity to individually tune each transistor to its minimum noise level separately.

All those transistors then will only deliver the current $I_{1,CLP}$ so when it comes to the FL-Phase an extra transistor is needed. This transistor can provide the additional current $I_{1,e}$ which represents the error made during the estimation.

The advantage then is that the transconductance of this latter current source can be smaller than the g_m of the transistor which takes care of I_1 completely (as formerly discussed). A smaller g_m means that the noise of the preceding stages will be suppressed in comparison with the former g_m.

This principle is an advantage of integrating the estimator with the oscillator in comparison with the external frequency estimator.

5.2.2 Binary coded current source

One possibility to implement such a current source is to design $N + 1$ different sources which all have their specific amount of output current. Source number

[3]Determination of I_1 during RL-Phase

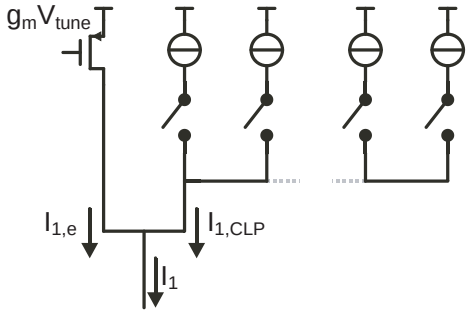


Figure 5.12: Semi-discrete current source I_1

n will then have $I_n = I_{max}/x^n$ as output current and every time a division by x is needed source n will be turned off and source $n+1$ turned on. This is drawn in Figure 5.12.

The smallest current source has to provide $1,67\mu A$ which is implementable. Though the drawback is that this setup requires a precise switching device in order to alternate the different sources. If the overlap in Figure 5.13 is t_e seconds

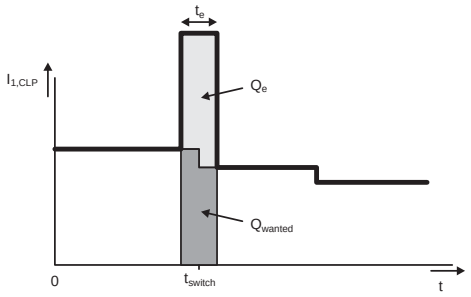


Figure 5.13: Error in binary coded current source during switching

long, then the error can be calculated as follows

$$\begin{aligned} f_e &= \frac{I_1}{\Delta V_2 C_2 - Q_e} - \frac{I_1}{\Delta V_2 C_2} \\ &= \frac{Q_e}{(\Delta V_2 C_2 - Q_e) \cdot \Delta V_2 C_2} \cdot \frac{I_{max}}{x^n} \end{aligned} \tag{5.13}$$

with Q_e the amount of charge which represents the error made. This Q_e is the amount of charge provided ($Q_{privided}$) within the period of t_e, minus the amount of charge (Q_{wanted}) which should be provided within this period.

The amount of charge which should be provided is equal to[4]

$$\begin{aligned} Q_{wanted} &= \frac{1}{2} t_e \cdot \frac{I_{max}}{x^n} + \frac{1}{2} t_e \cdot \frac{I_{max}}{x^{n+1}} \\ &= \frac{I_{max}}{x^n} \cdot \frac{x+1}{x} \cdot \frac{1}{2} t_e \end{aligned} \tag{5.14}$$

[4]Assuming t_e to be equally divided around the switching moment t_{switch}

while due to an error for example two current sources could be switched on simultaneously providing

$$\begin{aligned} Q_{provided} &= (\frac{I_{max}}{x^n} + \frac{I_{max}}{x^{n+1}}) \cdot t_e \\ &= \frac{I_{max}}{x^n} \cdot \frac{x+1}{x} \cdot t_e \end{aligned} \tag{5.15}$$

The error made is calculated as follows

$$\begin{aligned} Q_e &= Q_{provided} - Q_{wanted} \\ &= I_{max} \cdot \frac{x+1}{2x^{n+1}} \cdot t_e \end{aligned} \tag{5.16}$$

Evaluating these relations[5] reveals that for $n = 0$ the value of $Q_e = 16fJ$ and that turns into a maximum difference $f_e = 77kHz$. f_e decreases as n increases, which can be intuitively understood since t_{period} increases with n and so the constant error t_e has less effect.

5.2.3 Thermometer coded current source

Another principle is to cumulate the different currents and represent $I_{1,CLP}$ as their sum. In this way the overlap due to timing errors is reduced significantly (see Figure 5.14).

$$Q_{provided} = \frac{I_{max}}{x^n} \cdot t_e \tag{5.17}$$

so Q_e is equal to

$$\begin{aligned} Q_e &= Q_{provided} - Q_{wanted} \\ &= I_{max} \frac{x-1}{2x^{n+1}} \cdot t_e \end{aligned} \tag{5.18}$$

Evaluating this relation with Equation 5.13 results in a maximum error of 11.5kHz. The smallest current source has to provide $596nA$ according to A.7 which is implementable as well.

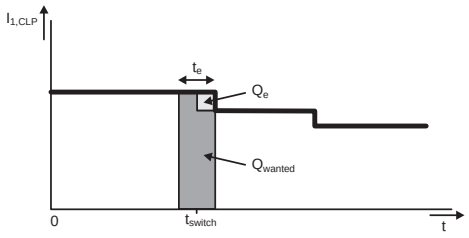


Figure 5.14: Reduced error in thermometer coded current source

[5]Assuming $\Delta V_3 = 800mV$, $t_e = 1ns$ and $I_{max} = 18,8\mu A$

5.2.4 Noise

One could wonder whether splitting $I_{1,CLP}$ in multiple current sources would raise the thermal noise contribution. So let one assume every transistor to be tuned to its equivalent noise resistance R_n according to

$$R_n \approx \frac{V_{DS,SAT}}{I_D} \tag{5.19}$$

and so

$$\begin{aligned} \overline{i_n^2} &= 4kT \cdot \frac{1}{R_n} \\ &\approx 4kT \cdot \frac{I_D}{V_{DS,SAT}} \end{aligned} \tag{5.20}$$

If all transistors are designed such that $V_{DS,SAT}$ is equal for each transistor then the thermal noise is calculated as follows

$$\overline{I_n^2} \approx \frac{4kT}{V_{DS,SAT}} \cdot I_{1,total} \tag{5.21}$$

$$\approx \frac{4kT}{V_{DS,SAT}} \cdot \sum_k I_{1,k} \tag{5.22}$$

in which $I_{1,total} = \sum_k I_{1,k}$. This means that the amount of thermal noise will remain the same if $V_{DS,SAT}$ is equal for every transistor.

The transistors could be degenerated in order to lower their noise contribution as originally done in the oscillator. If the resistor is big enough ($1/R_{deg} \ll g_m$) its noise behavior will dominate the thermal noise of the transistor. In practice the resistance should be chosen such that the transistor is out of headroom, meaning that $V_{R_{deg}}$ is maximized such that the transistor is just in saturation.

In Figure 5.15 such a degenerated source is drawn. Assuming node V_+ to be constant at $400mV$ one can examine what would be the necessary voltage across the resistor in order to bring the transistor on the edge of triode and saturation.

Evaluating this[6] in ProMOST reveals that $V_{DS,SAT} \approx 200mV$ leaving the voltage over the resistor to be equal to $600mV$.

If one would like to degenerate all the transistors of the discrete part of I_1 this can result in enormous resistors. For example, if one wants to degenerate a transistor which has to provide $1.67\mu A$ one needs a resistor of $360k\Omega$. A $596nA$ current source should contain a resistor of $1M\Omega$.

These values are very large and require a lot of chip area in order to realize them reliably. Recall that one needs about $1 + N = 1 + 8$ degenerated sources which vary between $24k\Omega$ and $360k\Omega$ in case of the binary coded current source and $91k\Omega$ and $1M\Omega$ in case of the thermometer coded current sources.

So with respect to the chip area needed for those resistors one could choose the binary coded current source. However one could get rid of the resistors. In

[6] The transistor is a double-gate PMOS with $V_{TH} \approx 400mV$ and the gate is connected to ground

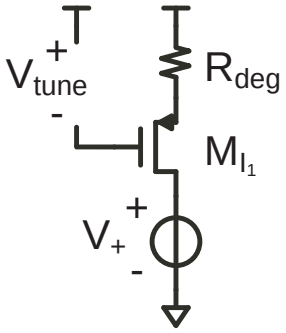


Figure 5.15: Degenerated current source

the original oscillator degeneration was used to linearize as well, as can be seen in Figure 4.3 compared to Figure 4.2. But in a discrete current source linearity is not an issue anymore, as the different values of the current sources can be designed as desired. They only have to be switched *on* or *off*.

So one can design the transistors such that their $V_{DS,SAT}$ is equal to ΔV_1. The point is then to get rid of the flicker noise which in the degenerated current source was dominated by the resistor. This flicker noise is usually modeled as a voltage source in series with the gate [8]

$$\overline{V_n^2} = \frac{K}{C_{ox}WL} \cdot \frac{1}{f} \tag{5.23}$$

Calculating the flicker noise at the drain current this will be

$$\overline{I_n^2} = \overline{V_n^2} \cdot g_m^2 \tag{5.24}$$

$$= \mu_p^2 C_{ox} K \frac{W}{L^3} \cdot \frac{1}{f} \cdot V_{GT}^2 \tag{5.25}$$

meaning that L should be increased while W is left to determine g_m. Figure 5.16 contains simulation results and reveals that indeed the noise of a single transistor can be similar to that of a degenerated version. $759fA/\sqrt{Hz}$ and $649fA/\sqrt{Hz}$ respectively[7].

As such one could either choose a binary coded current source or a thermometer coded current source. Since flicker noise reduction requires relative large transistors, mismatch will not be a problem in the accuracy of the current values. A 596nA source is expected to show less than a 10nA standard deviation due to mismatch according to ProMOST, which is negligible.

Since a thermometer coded current source is expected to be more accurate during switching it is better to choose a thermometer coded current source.

5.3 Small estimation step

In order to perform the small estimation steps according to Figure 5.17 an extra current source[8] (I_4) or an extra switched capacitor (C_4) can be added.

[7] Simulations done with $I_1 \approx 24\mu A$

[8] The source should be numbered 3 but has number 4 in order not to introduce extra confusions with C_1 and ΔV_3

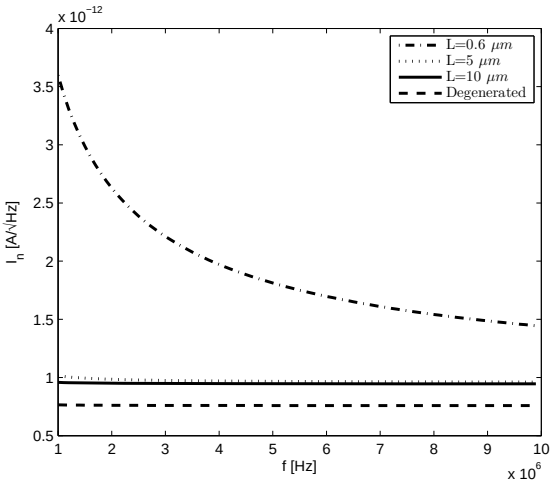


Figure 5.16: I_1 output referred noise results for non-degenerated PMOS transistors

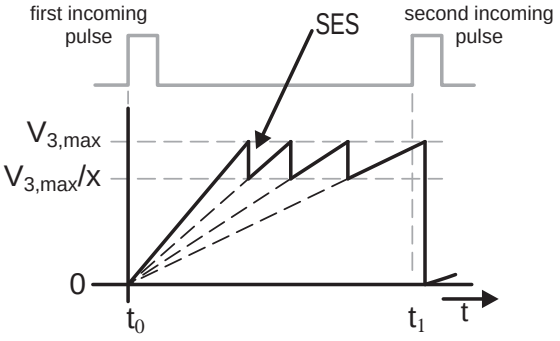


Figure 5.17: Estimation principle

5.3.1 I_4 versus C_4

Figure 5.18 shows option with an extra current source. In this case an extra current source is placed parallel with the switched capacitor C_2. Current source I_2 should then be variable (two sources). Despite the simplicity of this solution the drawback is that the amount of charge, in C_2, to be subtracted from C_1 will vary with the accuracy of the current source I_4. This accuracy depends on timing (on/of switching of the source determines the charge packet), mismatch and noise .

Another issue is the fact that ΔV_2 might vary with $V_{GS,M1}$. This means that $V_{3,max}$ of Figure 5.17 is not known exactly. This means that the exact value of the charge packets is unknown[9] as well. In order to keep the principle of Figure 5.2 the value of I_4 must be calibrated.

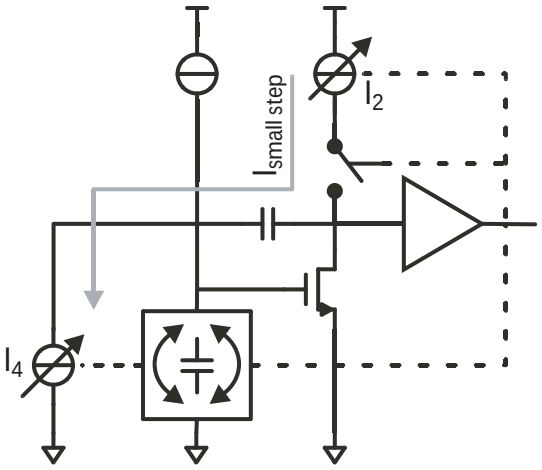


Figure 5.18: Small estimation step performed by extra current source(s)

Figure 5.19 shows a configuration which provides an automatically calibrated amount of charge to be subtracted from C_1. Since the circuitry already exists (by means of C_2), this only has to be copied and changed with a proper value for C_4. The accuracy of the charge packet now depends on mismatch and noise. Timing of switching does not affect the charge packet as it is fixed in relation to the charge packet of C_2 according to

$$C_4 = \frac{x-1}{x} \cdot C_2 \tag{5.26}$$

With C_2 equal to $2.5pF$ and $x = 1.35$ the value of C_4 should be $648fF$, which is feasible.

Chosen is to implement C_4 instead of I_4, because of its automatic calibration of the charge packet.

5.3.2 Discharge time

C_4 brings along a discharge time in combination with the impedance of M_1: $\tau_{SES} = C_4/g_{m_1}$. In Figure 5.20 the RC-time of the small estimation step is

[9]Though simulations can provide a good approaches

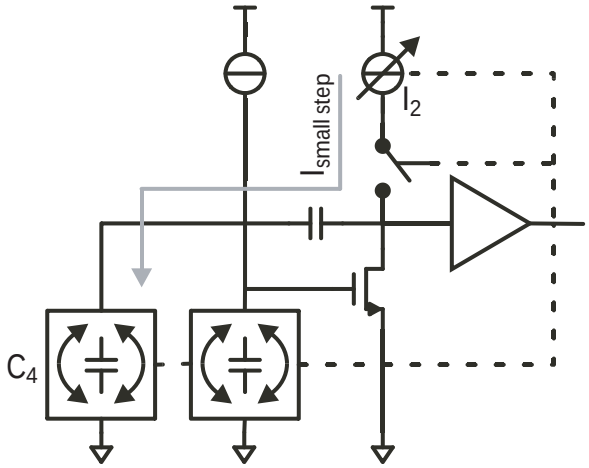


Figure 5.19: Small estimation step performed by an extra switched capacitor

depicted. This time may be longer than the time between two small estimation steps ($T_{allowed}$). Fortunately the RC-time appears be less than $T_{allowed}$. The

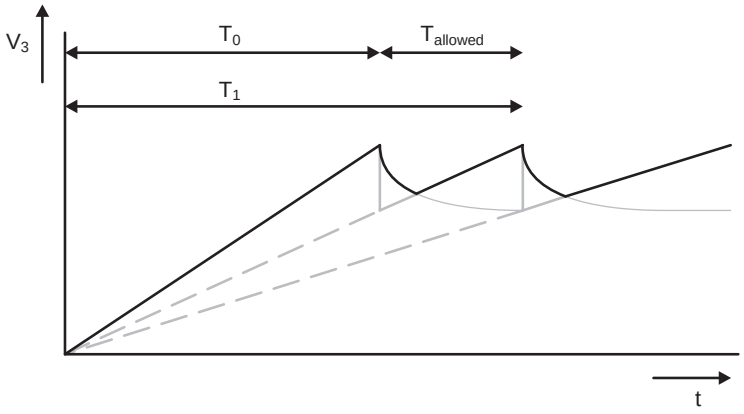


Figure 5.20: RC-time in small estimation step

allowed time is calculated as follows

$$T_{allowed} = T_1 - T_0 \tag{5.27}$$

with $T_0 = 2\pi/\omega_{max}$ and $T_1 = x \cdot T_0$, Equation 5.27 turns into

$$\begin{aligned} T_{allowed} &= T_1 - T_0 \\ &= T_0 \cdot (x-1) \\ &= \frac{2\pi}{\omega_{max}} \cdot (x-1) \end{aligned} \tag{5.28}$$

The time necessary to discharge is calculated as follows

$$\begin{aligned} T_{necessary} &= \tau_{SES} \\ &= \frac{C_4}{g_{m_1}} \\ &= \frac{C_2}{g_{m_1}} \frac{x-1}{x} \end{aligned} \tag{5.29}$$

The number of times that $T_{necessary}$ will fit in $T_{allowed}$ will be called γ and is equal to

$$\begin{aligned} \gamma &= \frac{T_{allowed}}{T_{necessary}} \\ &= \frac{T_0(x-1)}{C_2(x-1)/(g_{m_1}x)} \\ &= \frac{T_0}{C_2/g_{m_1}} \cdot x \end{aligned} \tag{5.30}$$

meaning that if x increases γ will increase as well. The limit[10] of γ for $x \downarrow 1$ is

$$\lim_{x \downarrow 1} \gamma = 18 \tag{5.31}$$

meaning that the number of RC-times will fit 18 times or more in $T_{allowed}$ for every $x > 1$. So, with respect to RC-times any $x > 1$ could be chosen. This can be seen as well in Figure 5.21.

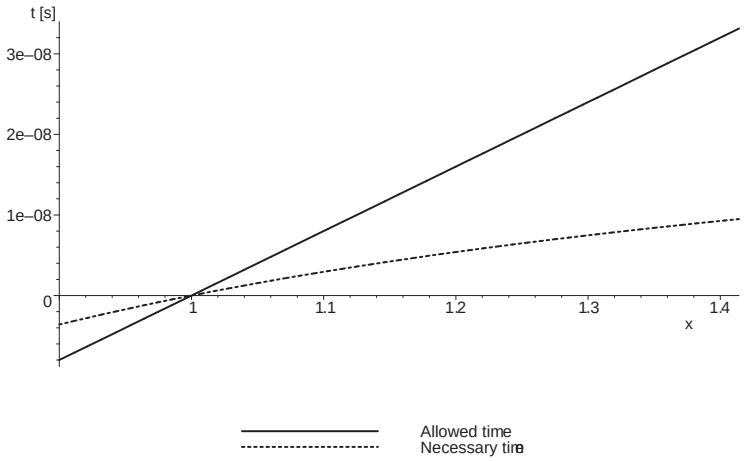


Figure 5.21: $T_{allowed}$ is always larger than $T_{necesarry}$ for $x > 1$

[10] With $\omega_{max} = 2\pi \cdot 10MHz$, $C_2 = 2.5pF$ and $g_{m_1} \approx 450\mu S$

5.3.3 Isolation of the switched capacitor

The reader could wonder whether C_2 should be isolated if C_4 is switched around. With respect to RC-time this is not necessary, according to calculations.

$$\begin{aligned} C_{total} &= C_2 + C_4 \\ &= C_2 + C_2 \cdot \frac{x-1}{x} \\ &= C_2 \cdot \frac{2x-1}{x} \end{aligned} \tag{5.32}$$

and so

$$\begin{aligned} \gamma &= \frac{T_{allowed}}{T_{necessary}} \\ &= \frac{T_0(x-1)}{C_{total}/g_{m_1}} \\ &= \frac{g_{m_1}T_0}{C_2} \cdot \frac{x(x-1)}{2x-1} \end{aligned} \tag{5.33}$$

With the same values as with the former calculation and $x = 1.35$ it appears that $\gamma \approx 5$. Again γ increases with x for every $x > 1$.

However if an ML-Phase is to be implemented one would like to take a sample of V_3 as a second pulse comes in. This sample should represent the error that was made during the rough estimation, see Figure 5.22. There are two

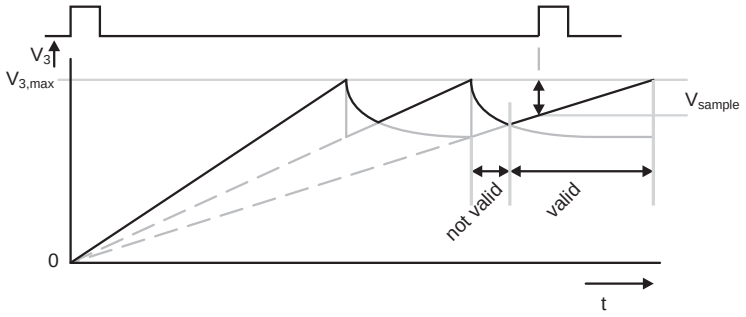


Figure 5.22: Time slots in which V_{sample} for the ML-Phase is valid or not

time slots in which such a sample is valid or not. If the small estimation step is being performed, a sample for the ML-Phase would not be proportional to ϕ_{out}. Discharging of C_1 should be done as quick as possible in order to get a valid sample.

Therefore the RC-time should be as short as possible and that is why C_2 is chosen to be isolated during a small estimation step. The same will be done with C_4 during the FL-Phase.

5.3.4 Constant ΔV_2 during the RL-Phase

ΔV_2 appears to be variable with I_1 through V_{GS,M_1} (Equation 4.2).

$$f = \frac{I_{max}}{(C_2 + C_{GS})\Delta V_2} \cdot \frac{1}{x^n} \tag{5.34}$$

Equation 5.34 states that the frequency is influenced by $\Delta V_2 = 2 \cdot V_{GS,M_1}$. The result, with respect to the RL-Phase, can be seen in Figure 5.23b. The result is that more small estimation steps are needed to cover all frequencies between ω'_{min} and ω'_{max}.

This can be seen as well in Figure 5.24, where different frequency ranges have been calculated according to Equation 5.34 for either a fixed ΔV_2 and a varying ΔV_2 according to Equation 4.2.

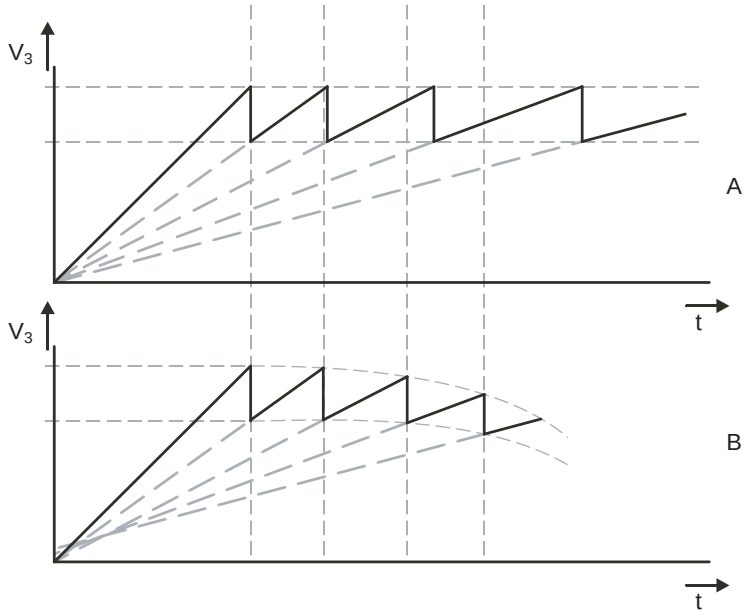


Figure 5.23: Estimation with either a fixed (a) and variable (b) ΔV_2

As can be seen indeed 8 steps/iterations[11] are needed to go down from ω'_{max} to ω'_{min} if ΔV_2 is fixed to $2 \cdot V_{GS,M_1} = 2 \cdot 400mV$.

The number of steps needed cover the same frequency range, with $\Delta V_2 = 2 \cdot V_{GS,M_1}$ as function of I_1, is about $N = 9$ to 10 steps. This means that the rough frequencies, ω_n, are not optimally chosen with respect to the Equations 5.10 and 5.11.

This will not be a very big problem, but as can be seen in Figure 5.23a all current slopes point to the origin of the plot. This is not the case in but Figure 5.23b. Due to the choice of C_4 instead of I_4 the factor x remains constant.

It could also be that $T_{allowed}$ becomes to short, though no calculations have been done.

Chosen is to place rough frequencies, ω_n, such that they are optimal with respect to the period jitter (Equations 5.10 and 5.11). The implemented solution to keep ΔV_2 constant, during the RL-Phase, is according to Figure 5.25.

As one of the switches 1_a, 1_b or 1_c is switched, its corresponding switch 2_a, 2_b or 2_c will be switched as well. If, for instance, only switches 1_a and 2_a are closed, value of $V_{GS,M_{1a}}$ would be $\approx 400mV$. Now switch 1_b will be closed as well and so switch 2_b. M_{1b} is dimensioned such that, with the current from I_{1b}

[11] As calculated with Equation 5.10, 5.11 and 5.12

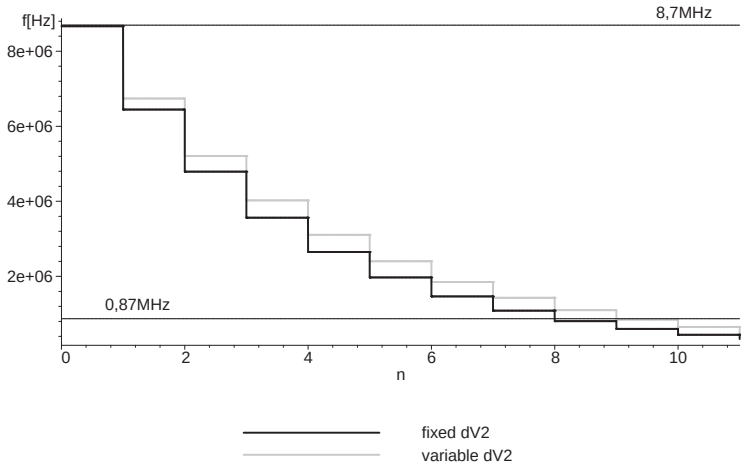


Figure 5.24: Evaluation of either a fixed and a variable ΔV_2

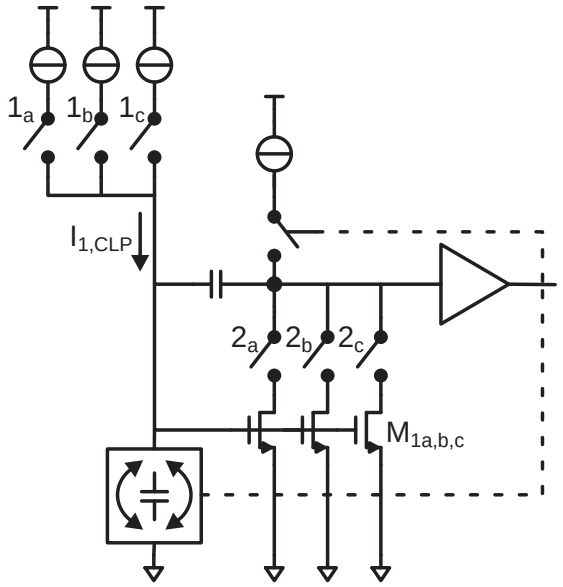


Figure 5.25: Constant $\Delta V_2 = 2 \cdot V_{GS,M_1}$ during RL-Phase

only, it has a $V_{GS,M_{1b}} \approx 400mV$. In this way the ΔV_2 is kept constant during the RL-Phase.

In the FL-Phase no steps are taken to keep ΔV_2 constant. The feedback loop will adjust finally to a proper value of I_e.

This constant ΔV_2 appears to have an extra advantage regarding the phase noise of the switched capacitor oscillator itself. The advantage is not significant but only mentioned here.

The phase noise is partly due to a varying charge packet to be subtracted from C_1. The deviation of the charge in C_1 is evaluated according to

$$\sigma^2_{Q_{C_1}} = 2kT \cdot \frac{Q_{in}}{\Delta V_{eff}} \tag{5.35}$$

in which[12]

$$\Delta V_{eff} = \frac{\Delta V_1 \cdot \Delta V_2}{\Delta V_1 + \Delta V_2} \tag{5.36}$$

So $\sigma^2_{Q_{C_1}}$ is dependent on V_{GS,M_1} through ΔV_{eff}. However Q_{in} varies as well with $V_{GS,M1}$ leaving the FoM (Equation 3.1) of the original oscillator almost constant. As can be seen in Figure 5.26 for V_{GS,M_1} around V_{TH,M_1} the FoM is

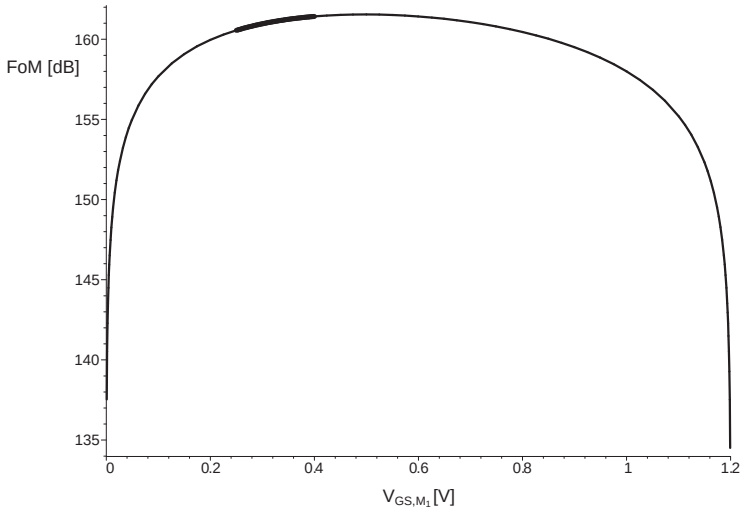


Figure 5.26: FoM calculation of the original oscillator

almost constant (thick part of the curve varies with 0.8dB over the thick part).

As V_{GS,M_1} will always be around $\approx 400mV$ during RL-Phase and FL-Phase and for all frequencies between $1MHz$ to $10MHz$, the FoM will be more constant than with a variable ΔV_2.

5.4 Estimation options

One could wonder whether the estimation curve of Figure 2.7 is the best solution to perform this estimation. Several possibilities are drawn in Figure 5.27.

[12]In this case again the ΔV_1 and ΔV_2 are regarded as *allowed* voltage swings

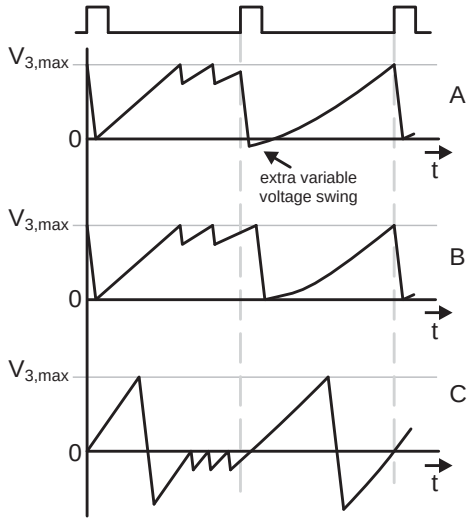


Figure 5.27: Different estimation options

In case of Figure 5.27 *a* the system will immediately discharge as the second incoming pulse arrives. As discussed previously as this pulse arrives the voltage difference between V_3 and $V_{3,max}$ provides information about ϕ_e to perform a ML-Phase. If one indeed immediately discharges the C_1 the sample could be affected.

The voltage swing ΔV_3 is variable as well and varies with $V_{3,max} - V_3$. Unfortunately though this will only happen during the RL-Phase and not during the FL-Phase, which is a waste of the voltage headroom needed to make the voltage swing of ΔV_3 possible.

Figure *b* deals with this. As the second pulse arrives the system keeps charging C_1 to $V_{3,max}$ and then performs the large voltage swing. As such V_3 will always stay within fixed (possibly optimal) boundaries.

Moreover, there is more time to sample V_3 for the ML-Phase in comparison with curve of Figure *a*. However when the second pulse arrives exactly when V_3 reaches $V_{3,max}$, there still remains a problem.

Figure *c* deals with those issues. It has an extra reference level at $0V$ for the small iteration steps. This makes it more easy to take a sample from V_3 for the ML-Phase, because as the second pulse arrives C_1 will go through the reference level of $0V$ and go to $V_{3,max}$. The sample can now be taken even if the second pulse arrives at $V_3 = 0V$. ϕ_e is then 0 rad as well and the system/PLL is in phase lock.

In the FL-Phase the system will try to tune I_e such that $V_3 = 0V$ every time a incoming pulse arrives. If $V_3 = 0V$ every time a incoming pulse arrives the system/PLL is called in phase lock.

Moreover both Figures *a* and *b* are implementations with a pre-charged capacitor which will be discharged as the first pulse arrives. Keeping the capacitor charged in the initialization phase awaiting an incoming pulse requires more energy in comparison with keeping it discharged.

5.5 Feedback

As can be seen in Figure 5.28 a PLL has a feedback loop in order to compare the phase of the incoming and the output signal.

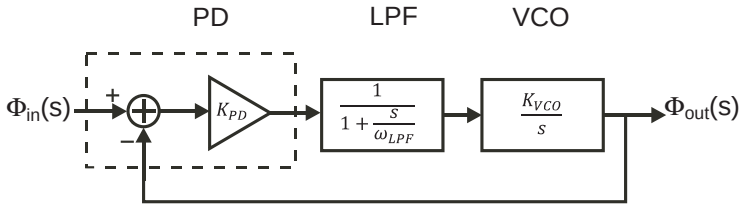


Figure 5.28: Typical PLL (Type I)

In order to acquire the phase information every time the incoming signal has an up-going flank the voltage across C_1 is being sampled. If V_3 is zero both ϕ_{in} and ϕ_{out} are equal.

This voltage V_3 can be sampled with a sample capacitor C_{sample}. This capacitor then contains an amount of charge which is representative for V_{sample} and thus for the phase error ϕ_e.

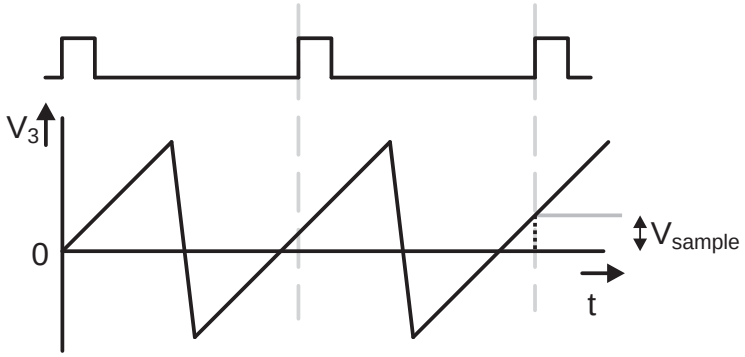


Figure 5.29: Acquirement of a representation of ϕ_e through V_{sample}

5.5.1 PLL type I or II

In case of a data signal to be received (with clock and data recovery for instance), typically a preamble will be sent first in order to lock to the reference frequency of the upcoming data signal. As the PLL has minimized ϕ_e (being in 'lock'), the data signal can be provided to the PLL. The PLL then checks whether its output signal and the transitions of the data signal remain in phase.

However, it will occur that for various periods no rising edges of the incoming signal will be detected (in case of two or more zeros for example)[7]. Hence no feedback will be performed and the oscillator must remain oscillating at its frequency.

This means that an integrator in the feedback loop is chosen. As no feedback is performed V_{tune} can remain constant.

The open loop transfer function has already a integrator by means of K_{VCO}/s. An extra integrator would introduce two poles at the origin of the s-plane of the open loop transfer function. This is called a type II PLL.

Two poles on the imaginary axis will cause the PLL to oscillate. To avoid this a zero will be added.

5.5.2 Integrator

A typical problem with a charge pump is that, if the excess phase ϕ_e is very small, the charge pump will not be able to charge C_{CP} (Figure 1.5) properly. For instance if the current sources I_{CP} are implemented by a single transistor with a switch, their slew rate can become too large for the small value of ϕ_e, see Figure 5.30.

Suppose $\phi_e \leq \phi_0$ then the pulses (to activate the switches and equivalent to those of V_{PD} in Figure 1.2) will contain not enough energy to turn those switches on. So if $\phi_e \leq \phi_0$ the phase will not be adjusted and the PLL 'waits' until $\phi_e > \phi_0$. This will cause extra jitter [8].

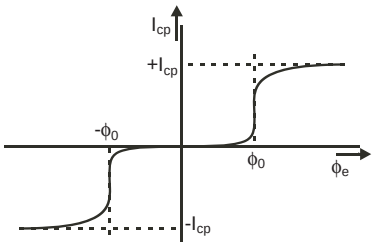


Figure 5.30: Deadzone in a general CPPLL

In some way the amount of charge in C_{sample} should be integrated and so distributed to C_{int}. All the information about the phase should be stored on this latter capacitor. The voltage over this capacitor C_{int} will be the voltage V_{tune}. A very basic integrator is displayed in Figure 5.31, two integrators are displayed.

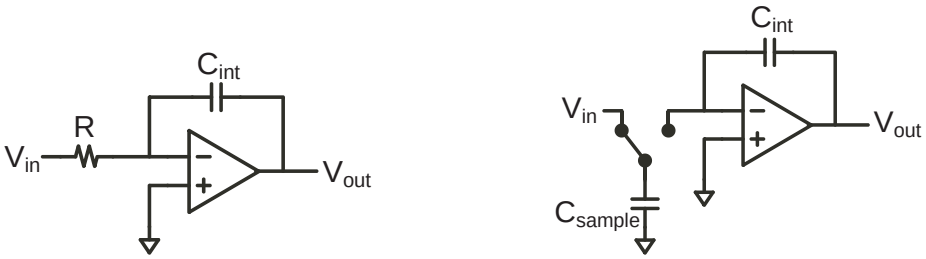


Figure 5.31: Integrators

The basic difference is that the resistor is replaced by a sampling capacitor. This capacitor transfers the incoming charge to the comparator, just like the resistor does continuously. If the sample frequency would be ω_{in} then the combination of the switch and C_{sample} could be regarded as a resistor with value $2\pi/(C_{sample}\omega_{in})$ [8].

Every time a sample is taken the amount of charge on C_{sample} is being redistributed actively towards C_{int}. A such the C_{int} becomes the C_{CP} of Figure 1.5 and C_{sample} with the comparator can be regarded as the charge pump. The amount of charge will be integrated with every incoming pulse. The voltage across C_{int} represents the tuning voltage with which the oscillator has to be tuned: V_{tune}.

The dead zone of Figure 5.30 was caused by the integrating switches which were switched too fast if ϕ_e became too small. Using the integrator of Figure 5.31 the minimum time to integrate will be equal to the shortest period of the incoming clock: $2\pi/\omega_{max} = 1/10MHz = 100ns$. By choosing for instance $R_{switch} \approx 100\Omega$ and $C_{sample} \approx 250fF$ the RC-time will be $\tau = 25ps$ which is 4000 times smaller than the shortest period of 100ns.

A problem of this integrator could be the charge injection of the sampling switches. But this is not investigated. Probably the error will be in the same range of the error due to switching in Figure 1.5.

5.5.3 Proposed integrator

The integrator of Figure 5.31 is a single-ended integrator. However V_3 in the oscillator is a differential voltage meaning that V_{in} of Figure 5.31 represents only V_- for instance and V_+ must still be subtracted. This is done in the configuration of Figure 5.32. As C_{sample} will be placed in parallel with C_1, a

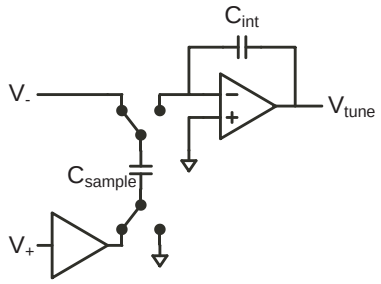


Figure 5.32: Differential to single-ended integrator

buffer is needed in order to keep the total capacity between V_+ and V_- equal to C_1.

Though only one buffer is needed to provide the amount of charge for C_{sample}, see Appendix B. Since V_- is kept in a feedback loop this node is already relatively stable. However, node V_+ is not, therefore a buffer between V_+ and C_{sample} is needed.

In order to overcome the need of a buffer, one could better realize a differential integrator as proposed in Figure 5.33. In this case V_+ and V_- are sampled with two different capacitors. As V_+ is almost constant over time, except when C_2 is switched, C_+ will barely affect the oscillator. C_- is still kept in the feedback loop of the oscillator's integrator, see Appendix B.

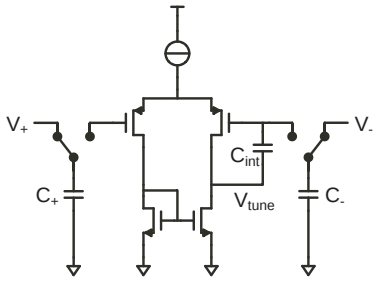


Figure 5.33: Differential to single-ended integrator to be implemented

5.5.4 Preset of V_{tune}

In the RL-Phase a rough value of I_1 is determined and in the FL-Phase an extra variable current source will take care of $I_{1,e}$. See Figure 5.34.

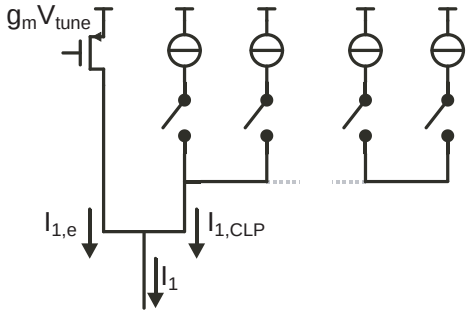


Figure 5.34: Semi-discrete current source I_1

However this tuning transistor should be turned off in the RL-Phase and turned on in the FL-phase. In order to be sure I_e is as close to zero as possible in the RL-Phase, V_{tune} should be a lot smaller than V_{TH}. But if the integrator in the FL-phase increases V_{tune} too slow, unwanted phase shift is introduced due to the fact that the tuning transistor will still be turned off for a several number of periods.

Besides that I_e is not turned on immediately when the FL-phase starts, there is another problem. Though the estimation was done such that most of the period jitter is covered in the several iteration steps, it might occur that $I_{1,CLP}$ too large.

Suppose an incoming signal has a period T_{in}, see Figure 5.35, which is just a little larger than T_{rough} for $n = 0$. However, due to period jitter the received signal has a period $T_{in} - \Delta T_{in}$ resulting to be just a little shorter than T_{rough}.

The system will now assume that T_{rough} is good and can only *add* a tuning current I_e, while, in fact, the chosen $I_{1,CLP}$ should be decreased.

In order to solve those two difficulties, recall that current source I_1 was a thermometer coded current source, from which every source contributes to the total current of $I_{1,CLP}$. If one current source within $I_{1,CLP}$ is switched off, $I_{1,CLP}$ is divided by x.

So, at the end of the RL-Phase ($I_{1,CLP}$ has been determined) one could

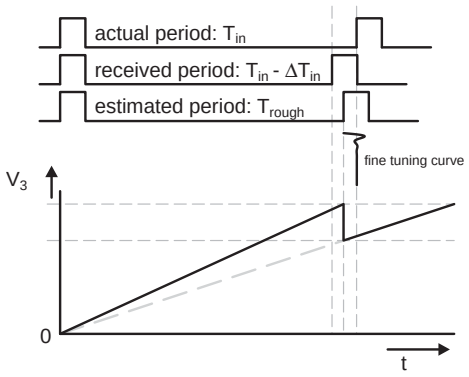


Figure 5.35: Estimation value $I_{1,CLP}$ appears to be too large

switch off one extra current source and map its value to the tuning transistor. M_tune is immediately turned on and I_1 can now also be less than $I_{1,CLP}$.

This is done in a straightforward way. All sources of $I_{1,CLP}$ were copied and through a current mirror and a switching network mapped to C_{int}. This can be seen in Figure 5.36.

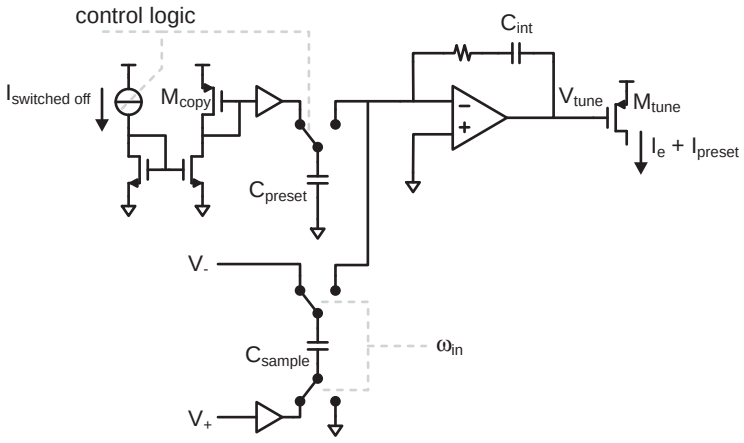


Figure 5.36: Presetting the integrator

M_{copy} and M_{tune} have the same dimensions so that the gate voltage of M_{copy} represents a proper V_{tune} to represent the current source which was switched off.

At this moment an extra current mirror is used in order to use the exact copy of the current sources of $I_{1,CLP}$. This mirror could be replaced by a NMOS current source with the same value for the current of the PMOS transistors.

Note that C_{int} and C_{preset} must also have the same values, in order to copy $V_{GS,M_{copy}}$ to V_{tune} with unity gain. A buffer is needed if C_{preset} appears to be too large to be charged to $V_{GS,M_{copy}}$ within a fraction of the period.

Another solution for these issues and avoids the need of a buffer, is that one of the current sources of $I_{1,CLP}$ is always turned on and could be used for I_e as

well. This is current source I_0 from Equation A.1.

So instead of an extra current source for I_e, as in Figure 5.34, now one of the RL-Phase current sources is used to provide I_e as well, see Figure 5.37. The

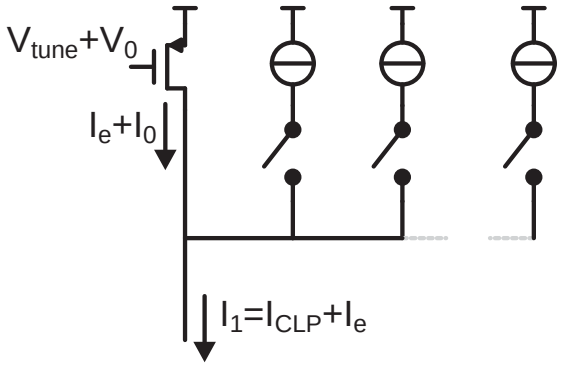


Figure 5.37: Alternative for presetting the integrator

reader might have noticed that indeed Figure 5.37 contains one current source less than Figure 5.34. One still has to think about a reliable way to set V_0.

Note that this latter option is not implemented yet. At this moment the configuration of Figure 5.36 is implemented.

5.5.5 Sample blocker

Since $\omega_{in} \neq \omega_{out}$ after the RL-Phase, ϕ_{out} 'walks away' from ϕ_{in}. This could cause an incoming pulse to occur while C_2 is switched and still settles.

This would not be a problem if C_1 was being discharged infinitely fast. But as can be seen in Figure 5.38 this is not the case due to the RC-time previously discussed. This will cause a misinterpretation of the phase, disorienting the feedback loop.

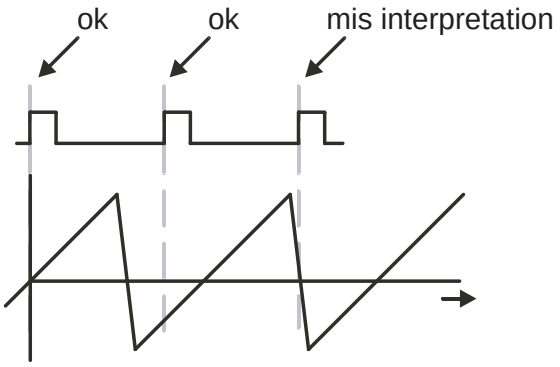


Figure 5.38: Misinterpretation of the excess phase

So in order to avoid a wrong sample, a blocker is built which avoids the feedback loop to sample during the discharging period. This will cause some extra phase shift, but is better than disorienting the feedback loop. The time

needed for discharging will take always the same amount of time τ. So during a fixed amount of time the circuit will not perform a sample. A sample blocker will disallow the circuit to take a sample.

5.5.6 Circuit

The overall circuit is depicted in Figure 5.39. This is without the preset circuit which is depicted separately again in Figure 5.40

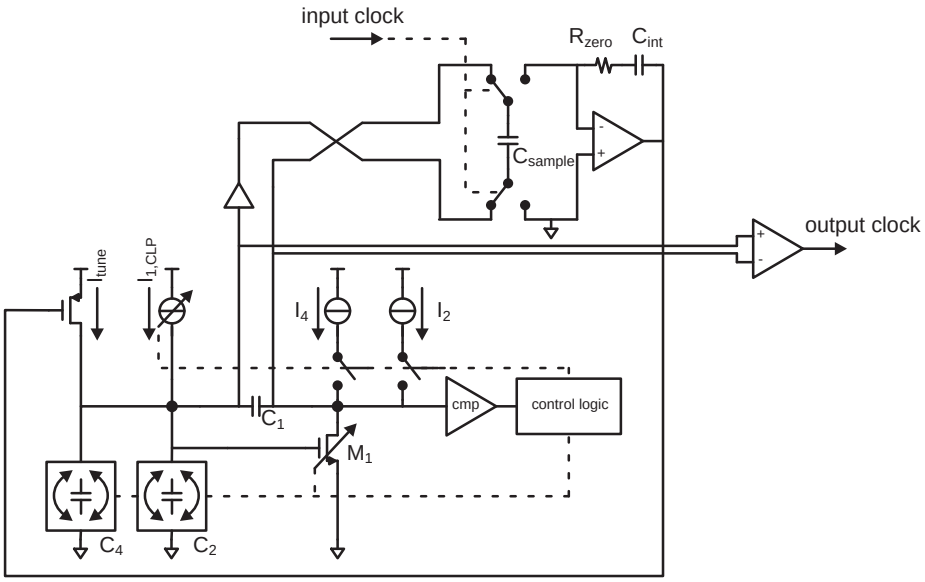


Figure 5.39: Circuit overview without the preset circuit and blocker

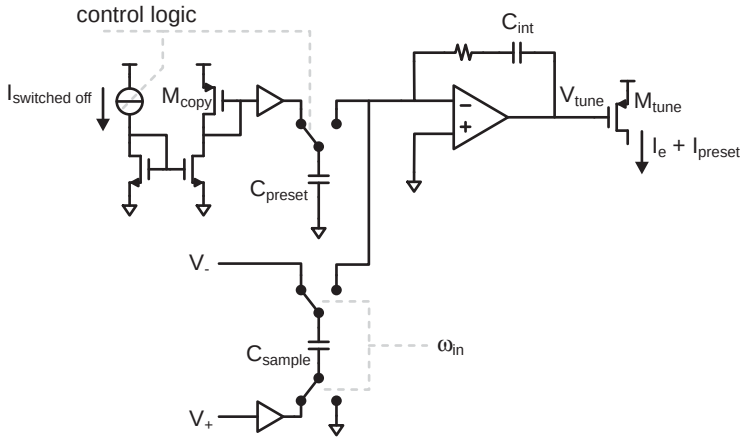


Figure 5.40: Presetting the integrator

Chapter 6

Model and simulation

This chapter describes the mathematical model of the PLL and the simulation results. This model should predict how the PLL will settle. Of main interest is then V_{tune} which is the output of the integrator and the input of the oscillator.

The whole PLL was built in Cadence CMOS065. Every component in the circuit is a CMOS065 model except for the integrator itself and the two buffers as described in Section 5.5. The latter two components are ideal building blocks.

6.1 Mathematical model

In order to derive a mathematical model of the PLL a graphical model according to Figure 6.1 is used. In the following subsections the different stages are analyzed.

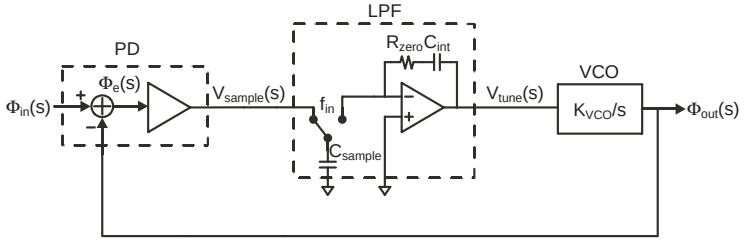


Figure 6.1: Graphical model of the PLL

6.1.1 Phase detector

The meaning of the phase detector's output, V_{sample}, can be retrieved with help of Figure 6.2. The phase detector should compare the input phase with the output phase by a subtraction.

$$\phi_e = \phi_{in} - \phi_{out} \tag{6.1}$$

This operation is always performed when an incoming pulse is detected, so $\phi_{in} \equiv 0$ or 1. ϕ_{out} is the phase of the output signal and is equal to the ratio

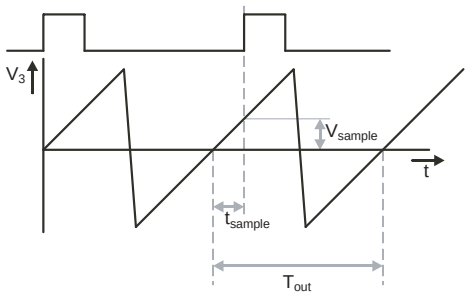


Figure 6.2: Graphical model of the PLL

between t_{sample} and the output signal's period T_{out}

$$\phi_{out} = \frac{t_{sample}}{T_{out}} \tag{6.2}$$

and with

$$t_{sample} = \frac{C_1 V_{sample}}{I_1} \tag{6.3}$$

$$T_{out} = \frac{(C_2 + C_{GS})\Delta V_2}{I_1} \tag{6.4}$$

this results in

$$\phi_{out} = \frac{C_2 + C_{GS}}{C_1} \cdot \frac{V_{sample}}{\Delta V_2} \tag{6.5}$$

$$\approx 1 \cdot \frac{V_{sample}}{\Delta V_2} \tag{6.6}$$

and so

$$\phi_e \approx 0 - \frac{V_{sample}}{\Delta V_2} \tag{6.7}$$

Now one can calculate the meaning of V_{sample}

$$V_{sample} \approx -\Delta V_2 \cdot \phi_e \tag{6.8}$$

6.1.2 Integrator

A PLL with an integrator as described in the previous chapter would contain two poles at the imaginary axis of the s-plane. This would result in an instable system and therefore a zero is added. The PLL of Figure 6.1 contains this a resistor called R_{zero} which creates this zero.

The transfer function of such an integrator is equal to

$$F_{int}(s) = \frac{V_{out}}{V_{in}}(s) = -\frac{1 + sC_{int}R_{zero}}{sC_{int}R} \tag{6.9}$$

Now replacing the resistor R by $(C_{sample}f_{in})^{-1}$ turns Equation 6.9 into

$$F_{int}(s)\frac{V_{out}}{V_{in}}(s) = -\frac{1 + sC_{int}R_{zero}}{sC_{int}} \cdot C_{sample}f_{in} \tag{6.10}$$

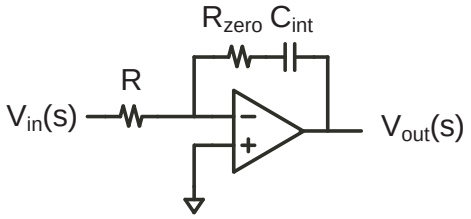


Figure 6.3: Integrator with a zero added

6.1.3 K_{VCO}

The oscillator's gain is equal to the derivative of the frequency with respect to its tuning variable. Note that $I_1 = I_{CLP} + I_e$ in which I_{CLP} is constant during fine locking phase.

$$\begin{aligned} f &= \frac{I_{CLP}}{C_2 \Delta V_2} + \frac{I_e}{C_2 \Delta V_2} \\ &\approx f_{CLP} + \frac{g_m}{C_2 \Delta V_2} \cdot V_{tune} \end{aligned} \tag{6.11}$$

$$\begin{aligned} K_{VCO} &= \frac{df}{dV_{tune}} \\ &= \frac{g_m}{C_2 \Delta V_2} \end{aligned} \tag{6.12}$$

Note that actually C_2 supposed to be $C_2 + C_{GS}$. Note as well that $V_{tune} = -V_{out}$ of the integrator. This is due to the fact that the current source I_1 is a PMOS device.

6.1.4 PLL dynamics

Now knowing all dynamics of every single stage one can calculate the overall behavior of the PLL with respect to input and output phases. The open loop transfer function is equal to

$$\begin{aligned} H_{OL}(s) = \frac{\Phi_{out}}{\Phi_{in}}(s) &= K_{PD} \cdot F_{int}(s) \cdot \frac{K_{VCO}}{s} \\ &= -\Delta V_2 \frac{1 + sC_{int}R_{zero}}{sC_{int}} \cdot C_{sample} f_{in} \frac{-g_m}{sC_2 \Delta V_2} \\ &= \frac{1 + sC_{int}R_{zero}}{s^2 C_2 C_{int}} \cdot C_{sample} g_m f_{in} \end{aligned} \tag{6.13}$$

The closed loop transfer function is then calculated as follows

$$\begin{aligned} H_{CL}(s) &= \frac{H_{OL}(s)}{1 + H_{OL}(s)} \\ &= \frac{(R_{zero}C_{int}s + 1) \cdot C_{sample} g_m f_{in}}{C_2 C_{int} s^2 + C_{int} C_{sample} g_m f_{in} R_{zero} s + C_{sample} g_m f_{in}} \end{aligned} \tag{6.14}$$

so

$$\omega_n = \sqrt{\frac{C_{sample} g_m f_{in}}{C_2 C_{int}}} \tag{6.15}$$

$$\zeta = \frac{R_{zero}}{2C_2} \cdot \sqrt{C_2 C_{int} C_{sample} g_m f_{in}} \tag{6.16}$$

If one wants to suppress the phase noise of the incoming signal as much as possible, ω_n should be chosen as low as possible. This implies that C_2 and C_{int} must be chosen large and C_{sample} and g_m as small as possible[1].

In order to calculate the settle time recall that the zero influences the absolute damping. As the complex poles have a real part value of $-\omega_n\zeta$ the transient step response has a factor $e^{-\omega_n\zeta \cdot t}$. The time needed to settle within a margin of 1% of its final value is equal to

$$t_{1\%} = \frac{1}{-\omega_n\zeta} \cdot \ln 0.01 \tag{6.17}$$

However, $t_{1\%}$ does not give a clear insight since it is an absolute value. A rather low frequent signal takes more time to settle. Only once a period a sample of the excess phase is taken and since every period is longer than at high frequencies, so in absolute settle time varies with ω_{in} . Dividing by the period of the incoming clock signal one calculates the number of periods needed to settle within 1% of its final value

$$n_{1\%} = \frac{f_{in}}{-\omega_n\zeta} \cdot \ln 0.01 \tag{6.18}$$

$$= -2 \cdot \frac{C_2}{R_{zero} C_{sample}} \frac{1}{g_m} \cdot \ln 0.01 \tag{6.19}$$

As can be seen the number of periods needed to settle within a certain range is independent of the incoming signal's frequency. Note that this is the number of iterations needed to get from $\Phi_{in}/\Phi_{out} = 0$ to $\Phi_{in}/\Phi_{out} = 1$. But in most cases the $\Phi_{in}/\Phi_{out} \neq 0$ after estimation. So in this case Equation 6.19 turns into

$$n_{1\%} \leq -2 \cdot \frac{C_2}{R_{zero} C_{sample}} \frac{1}{g_m} \cdot \ln 0,01 \tag{6.20}$$

6.2 Results

After all considerations simulations were performed. These were done in the CMOS065 process. During the implementation in the simulator values for different components were chosen. Note that the integrator is implemented with an ideal voltage controlled voltage source. Due to this, the simulations show a lot of spikes at the curve of V_{tune} when switches a switching. In the simulation results those spikes were suppressed mathematically, to clearly show V_{tune}. When an transistor model would be used those spikes are expected to be suppressed as well.

[1] In this chapter ω_{in} represents the natural frequency of the PLL

6.2.1 Component values

The simulations are done with a sample capacitor C_{sample} of $250fF$ and as such 10 times smaller than C_1. As seen in Equation 6.15, C_{sample} should be chosen as small as possible. The integration capacitor is chosen to be $C_{int} = 2.5pF$ large according to that same Equation 6.15. This value may be chosen larger as it would lower ω_n. In order to maintain $\zeta = \frac{1}{2}\sqrt{2}$ R_{zero} could then be chosen smaller.

The value of g_m was chosen arbitrarily, just able to provide more than enough current to represent one single current source of I_1 and I_e. The length was chosen $1\mu m$ in order to order to reduce the channel length modulation. The width of $1\mu m$ as well was chosen such that with a $V_{tune} = 1.2V$ the output current is equal to $I_e \approx 18\mu A$. The value of g_m then turns out to be $24\mu S$.

With help of Equation 6.16 and choosing $zeta = \frac{1}{2}\sqrt{2}$, which in the second order system equals a phase margin of 60 degrees, $R_{zero,opt}$ should equal $228k\Omega$ when $f_{in} = 6.67MHz$. $R_{zero,opt}$ can be calculated according to

$$R_{zero,opt} = \sqrt{\frac{2C_2}{C_{int}C_{sample}g_m}} \cdot \sqrt{\frac{1}{f_{in}}} \tag{6.21}$$

which is Equation 6.16 with $zeta = \frac{1}{2}\sqrt{2}$ evaluated. Within the frequency range of $\omega_{in} = 2\pi \cdot 1MHz$ up to $2\pi \cdot 10MHz$ this results in a variable $R_{zero,opt}$ of $183k\Omega$ for $10MHz$ to $577k\Omega$ for $1MHz$.

The value of ω_n appears to vary between $2\pi \cdot 153kHz$ and $2\pi \cdot 484kHz$ according to Equation 6.15. At $6.67MHz$ simulations show a $\omega_n = 2\pi \cdot 368kHz$ which differs $27kHz$ from the calculated frequency of $\omega_n = 2\pi \cdot 395kHz$.

The open loop bandwidth of the PLL is with these values for the components smaller than $\approx 0.08 \cdot \omega_{in}$. This means that a s-domain description of the PLL's behavior will give reliable results.

6.2.2 Plots

In Figure 6.4 one can see the estimator estimating the frequency of an incoming clock signal of $1.25MHz$. The reader can see that the slope of V_3 for the different values of I_1 have their origin in $0 - \Delta V_{2,bigstep}$. The small step seems to vary, but this is due to RC-time of discharging and different charge values of $I_{1,CLP}$. The $\Delta V_{2,smallstep}$ mark in the plot shows the actual small estimation step. Though one cannot read out the actual $\Delta V_{2,smallstep}$, the estimation is that indeed $x \approx 1.35$.

The first three estimation steps show a V_3 curve that obviously exceeds $0V$. This is due to the slew rate of the comparator which decides when a small step has to be made.

In Figure 6.5 one can see that the tuning current source takes over one of the sources of the thermometer current source. Immediately after is has been set, the fine tuning starts. Zooming in, one sees Figure 6.6, which shows the response of the mathematical model as well. In these plots R_{zero} was chosen $R_{zero} < R_{opt}$.

As stated before the mathematical model predicts a maximum number of periods needed tot arrive within 1% of the final value by Equation 6.20. However, in most cases the ratio Φ_{in}/Φ_{out} does not start from 0, but some where

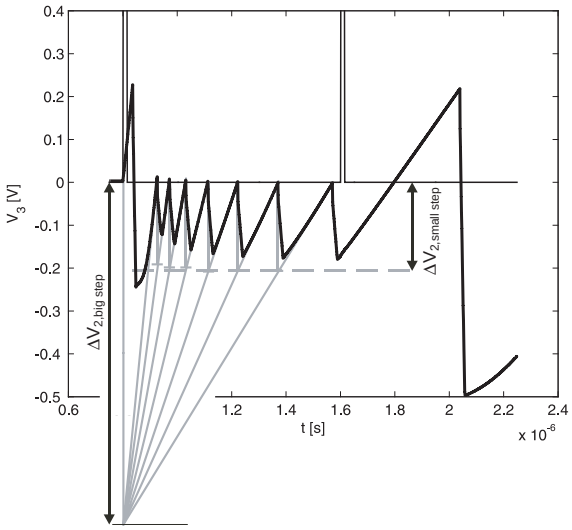


Figure 6.4: Estimating an incoming signal of $1.25MHz$

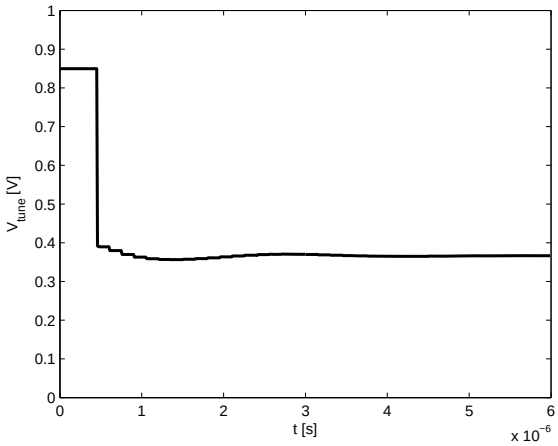


Figure 6.5: V_{tune} is set to represent a part of $I_{1,CLP}$ ($R_{zero} < R_{opt}$)

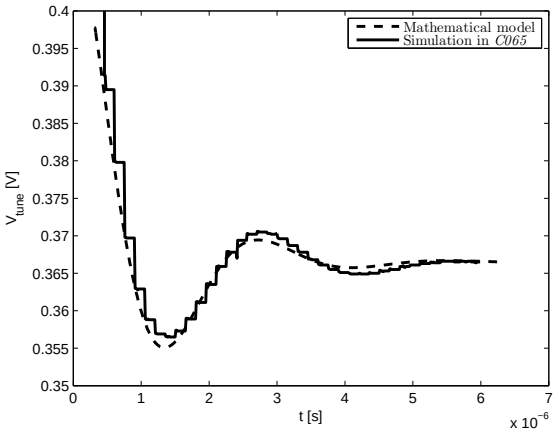


Figure 6.6: Verification of mathematical model through V_{tune} ($R_{zero} < R_{opt}$)

between 0 and 1. So the number of iterations will be smaller. This can be seen in Figure 6.7 in which two boundaries are depicted which represent the 1% margin of the final value ($R_{zero} = R_{opt}$).

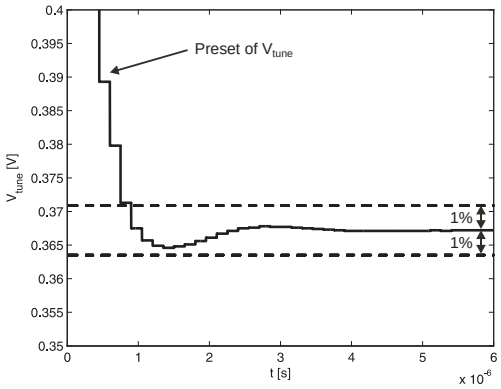


Figure 6.7: $n_{1\%} = 3$ samples needed ($R_{zero} = R_{opt}$)

In Figure 6.8 one can see that node V_+ indeed tries to remain at $V_{GS,M_1} \approx 400mV$. It depends the value of I_e how much it will exceed.

The reason that $V_{+,max}$ varies is that M_1 is designed such that $V_{GS,M_1} = constant$ within the RL-phase. But in the FL-phase M_1 is not compensating for I_e, so that is why V_{GS,M_1} must increase as I_e increases. If one looks carefully one might notice that the values for $V_{+,max}$ show the same slope as V_{tune} does (inverted though).

6.2.3 Process corners

Though different stages are still ideal components (integrator and the two buffers), the PLL has been simulated over all process corners. It performs normal if the NMOS transistors are fast and the PMOS transistors both slow and fast. But

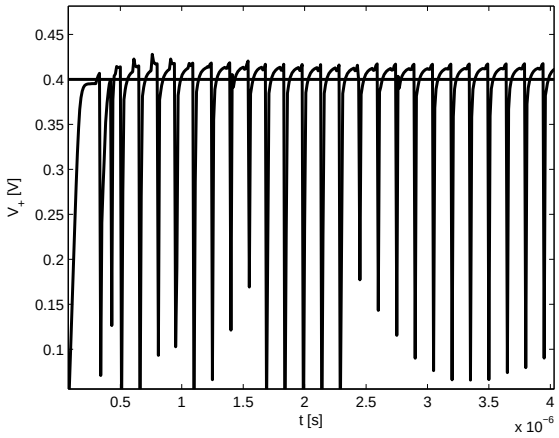


Figure 6.8: V_{GS,M_1}, and so V_+, stays around 400mV just before a capacitor is switched

if the NMOS transistors are in the slow corner the PLL fails. This has probably to do with timing in the control logic, though this has not been confirmed yet.

This timing issue should be focussed on, since the simulations of the failing process corners show similar results as with old simulations with the nominal model, where indeed timing in the control logic was a problem.

6.3 Improving

An ω_n that varies from $2\pi \cdot 153kHz$ to $2\pi \cdot 484kHz$ is rather high. As ω_n is equal to

$$\omega_n = \sqrt{\frac{C_{sample} g_m f_{in}}{C_2 C_{int}}}$$

ω_n can be decreased by increasing C_{int} which now has the value of $2.5pF$. Let one do some calculations at the frequency $\omega_{in} = 2\pi \cdot 6.67MHz$. The value of ω_n can be reduced to $2\pi \cdot 88kHz$ with $C_{int} = 50pF$ and to $2\pi \cdot 62kHz$ with $C_{int} = 100pF$. According to the equation for ζ the value of R_{zero} may be lowered to $R_{opt} = 36k\Omega$. However, $n_{1\%}$ will then be equal to 110 periods ($16\mu s$).

The reader might have noticed that $n_{1\%}$ does not vary with C_{int}. It decreases as R_{zero} increases. In fact, if R_{zero} is remained $228k\Omega$ and $C_{int} = 50pF$ the number of iterations in the FL-phase will be $n_{1\%} \leq 18$ which in absolute time is $18/\omega_{in} \sim 2.7\mu s$.

But a lager C_{int} implies a larger C_{preset} as well as $C_{int} = C_{preset}$. This means that C_{preset} needs to be $50pF$ as well. At this moment the switches to perform the preset of V_{tune} provide too much resistance to perform the preset properly. This should be adjusted to lower ω_n.

As in the current simulation it turns out that a large C_{int} and C_{init} will not preset V_{tune} properly a simulation with an exaggerated value of $R_{zero} = 1M\Omega$ was performed. $n_{1\%}$ is then calculated at 4 periods. Simulation results show a number of 2 periods after RL-phase, as seen in Figure 6.9. With respect to $n_{1\%}$ the value of R_{zero} should be as large as possible. Its value does not have to

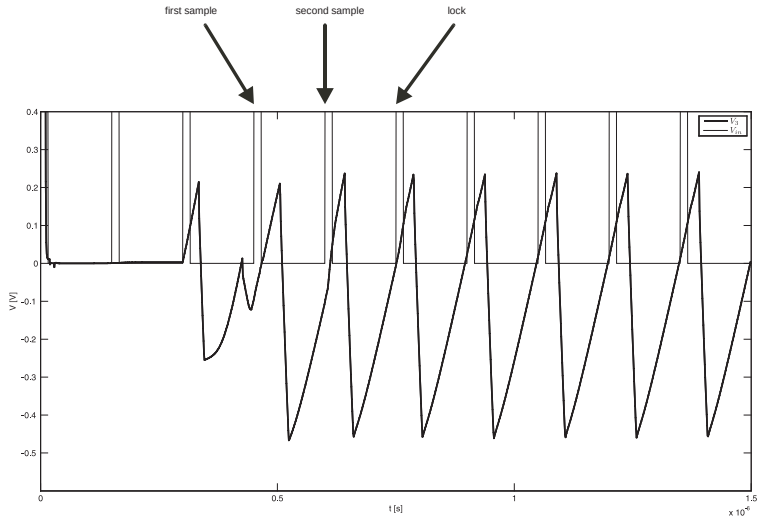


Figure 6.9: Phase lock within 2 periods

be precise, which means it can be implemented without accuracy, for example a transistor in triode. Note that no noise analysis is done. As such what R_{zero} does exactly with the noise a V_{tune} remains unknown.

Another possibility is to allow a high bandwidth during the RL-phase and FL-phase. After lock, one could enlarge $C_{int} = 2.5pF$ by placing an extra capacitor in parallel. Difficulty will be the fact that this extra capacitor has to be pre-charged in order to avoid charge redistribution.

At last one could consider a digital filter with a DAC as current source. Such a large value as C_{int} (in order to lower ω_n) would just be a number in the digital filter. This can be much smaller in chip area to implement.

A typical DAC has predefined output values for its output current. In order to provide the different values of I_{max}/x^n in the RL-Phase, the value of x could be set to $x = 2$. Or the resolution of the DAC could be chosen high enough such that the deviation from $x = 1.35$ becomes negligible. Possibly dithering could be considered, especially in the FL-Phase where arbitrary values of I_e are needed.

6.4 Summary

The model is verified with simulations, hence R_{opt} and ω_n can be predicted reliably. For the frequencies 1 to 10MHz the value of R_{opt} lies within the range of $183k\Omega \ldots 577k\Omega$ and ω_n varies from $153kHz \ldots 484kHz$.

If C_{int} is enlarged to $50pF$ the ω_n varies from $2\pi \cdot 34kHz$ till $2\pi \cdot 108kHz$. The value of R_{zero} should remain large, though nothing about phase noise is known yet.

A digital filter could be considered.

Chapter 7

Conclusions and recommendations

Several conclusions can be made. The linearity of the oscillator has been investigated as well as the possibility to realize a PLL with a frequency estimator. Afterwards some recommendations will be made.

The following questions were of interest at the beginning of the project.

1. What problems will be encountered to realize such a circuit?
2. How could the feedback loop be realized?
3. How to assure long term stability of the oscillator?

The first two questions have been answered and will be summarized in the next sections. The third remains for further research.

7.1 Linearity

An investigation of the constant K_{VCO} of the switched capacitor oscillator has been done. It appears that the $C_{GS,M1}$ which is in parallel with C_2 varies with the frequency. It appeared that $C_{GS,M1} \approx 0.2pF$ while $C_2 = 2.5pF$. As such the charge packet varies with the frequency as well, resulting in a behavior that is more linear than expected.

7.2 PLL with frequency estimator

This project was about the reduction of the settle time of a PLL. Such a reduction is not necessary in systems where data is send continuously. But in burst wise communication this settle time becomes significant.

The idea of first doing a frequency estimation before comparing ϕ_{in} and ϕ_{out} was expected to be a possible solution to reduce the settle time.

Simulations have shown that this is indeed is the case and is expected to be feasible in CMOS065. Almost everything was implemented in models with parasitics, except for the integrator and two voltage buffers. Given the circuit behavior, a suggestion for an implementation of these components was done.

A sample blocker has been implemented in order to avoid a misinterpretation of V_{sample} in the feedback loop. C_1 will not be discharged infinitely fast, but a RC-time is involved. During this discharging period a sample may not be taken as it is meaningless with respect to ϕ_e.

With respect to all process corners the PLL works except for the slow NMOS corners. This is probably due to timing in the control logic.

Hence, it can be concluded that a fast locking PLL with frequency estimator, as it is implemented according to the considerations of this report, *is expected to be* feasible.

The simulations run, show a settle time of 1+3 periods. One period for the estimation and three for the fine locking. This was done at an input frequency of $6.67MHz$ and as such requires $4 \cdot 150ns = 600ns$ to lock within a margin of 1%. However, with a rather high natural frequency of $153kHz$ up to $484kHz$ over the whole frequency range of $1MHz$ to $10MHz$.

A mathematical model was derived in order to get more insight in the system parameters. This model appeared to be plausible in comparison with the simulations.

With help of the model, adjustments could be considered. The integration capacitor C_{int} could be enlarged. This would lower the bandwidth of the PLL, while this does not affect the number of periods needed to settle.

The value of R_{zero} is calculated relatively large as well. $R_{zero} = R_{opt}$ can be decreased as well to a value of $39k\Omega$ by enlarging this C_{int}.

Although bandwidth and relative damping appear to be dependent on ω_{in}, the number of periods needed to settle is not. This could be useful to define a fixed preamble for all frequencies.

7.3 Recommendations

In order to reduce the settle time the ML-Phase could be introduced. At the end of the CL-phase a sample of V_3 could be taken. If the system assumes that the incoming pulse represents $T_{in} + \Delta T_{in}$ with $\Delta T_{in} = 0$ the preset of V_{tune} could be improved. This could refine the estimation and as such shorten the settle time.

The integrator proposed could be implemented and simulated as well as the alternative for the preset of V_{tune}. And to make the PLL reliable one should make it work over all process corners.

In order to realize a low bandwidth, one could consider a variable C_{int}, which could be small during settling and large after settling. As such the system could settle rapidly with a high bandwidth and afterwards lower the bandwidth in order to reduce phase noise.

As C_{int} can become very large in order to lower the bandwidth, it is recommended to implement the filter digitally. A current providing DAC could replace I_1.

No noise calculations were done. The amount of noise at the input of the oscillator is needs to be investigated. And, though effort has been made to preserve the phase noise of the oscillator itself, extra noise could have been introduced as the output signal is determined as V_3 crosses zero. If the PLL is

in lock, the samples to determine the phase will be taken as V_3 crosses zero as well. This should be further investigated.

Bibliography

[1] Geraedts, P.F.J. "Realization of an Integrated Clock Cleaner Circuit," ICD - University of Twente, Report nr:067.3131, Dec. 2005

[2] P.F.J. Geraedts, E. van Tuijl, B. Nauta et al. "A 9O μ W 12MHz Relaxation Oscillator with a -162dB FoM," IEEE ISSCC, pp.348-618, Feb. 2008

[3] M. J. Underhill, "The Adiabatic Anti-Jitter Circuit," IEEE Transactions on ultrasonics, ferroelectrics and frequency control, vol. 48, no. 3, May 2001

[4] M. Banu, A.E. Dunlop, Y. Ota et al., "Design Techniques for Non-Oversampled, Instantaneously Locked Clock and Data Recovery Circuits - An Overview," Digest of the LEOS Summer Topical Meetings pp.12-13, Aug. 1995

[5] G. Yan, F. Wennan, J. Lijiu et al., "A fast locking charge-pump PLL with adaptive bandwidth," ASIC 2005. ASICON 2005. 6th International Conference On Vol. 1, pp.383-386, Oct. 2005

[6] R. Navid, T.H. Lee, R.W. Dutton, "Minimum Achievable Phase Noise of RC Oscillators," IEEE J. Solid-State Circuits, vol. 40, NO. 3, pp. 630-637, March 2005

[7] J. Watkinson, "The art of digital audio," Oxford Focal Press, ISBN 0-240-51320-7, 1994

[8] B. Razavi, "Design of Analog CMOS Integrated Circuits," McGraw-Hill, ISBN 0-07-118815-0, 2001

[9] B. Razavi, "RF Microelectronics," Prentice Hall, ISBN 0-13-887571-5, 1998

Appendix A

Thermometer coded current source

A certain current to be produced by N different current sources, which in total should provide I_{max} and with the summation of n sources should provide $I = I_{max}/x^n$, should be designed as follows.

The smallest current to be delivered is equal to[1]

$$I_0 = \frac{I_{max}}{x^N} \tag{A.1}$$

This source forms the basis of the total current to be provided. Every current needed to be provided is due to a summation of small current sources and I_0.

So, what should be the value of current source I_N in order to provide $I = I_{max}/x^{N-1}$? Note that I_N is the smallest current to be provided in the thermometer coded current source. The difference between the two currents I and I_0 is equal to I_N, hence

$$\begin{aligned} I_N &= I - I_0 \\ &= \left(\frac{1}{x^{N-1}} - \frac{1}{x^N}\right) \cdot I_{max} \\ &= \frac{x-1}{x^N} \cdot I_{max} \end{aligned} \tag{A.2}$$

This can be generalized to

$$I_n = \frac{x-1}{x^n} \cdot I_{max} \tag{A.3}$$

for every current source realized. As such the total current is equal to

$$I = I_0 + \sum_{n=1}^{N} I_n \tag{A.4}$$

[1] In the rest of the report I_0 would suggest that is represents the current belonging to I_{max} in the CL-Phase, the nomenclature in this Appendix is different. I_0 is the smallest current of I_{CLP}.

For what value of x is I_0 equal to I_N?

$$\begin{aligned} I_0 &= I_N \\ \frac{1}{x^N} \cdot I_{max} &= \frac{x-1}{x^N} \cdot I_{max} \\ x &= 2 \end{aligned}$$

What would be the smallest difference between two current sources? For I_0 can be equal to I_N, this question has no unambiguous answer. One could simply start calculating the difference between I_N and I_{N-1}, which should be the smallest current difference according to Equation A.3. But if x approaches 2 the distance between I_N and I_0 will be smaller than the difference between I_{N-1} and I_N. So, for what value of x is are both differences the same?

$$\begin{aligned} \Delta I_{0,N} &= I_0 - I_N \\ &= \frac{2-x}{x^N} \cdot I_{max} \qquad \text{(A.5)} \\ \Delta I_{N,N-1} &= \left(\frac{x-1}{x^{N-1}} - \frac{x-1}{x^N} \right) \cdot I_{max} \\ &= \frac{(x-1)^2}{x^N} \cdot I_{max} \qquad \text{(A.6)} \end{aligned}$$

so if one states that $\Delta I_{0,N} = \Delta I_{N,N-1}$ what would x be?

$$\begin{aligned} x &= \frac{1 \pm \sqrt{5}}{2} \\ &\approx -0,618 \text{ and } 1,618 \end{aligned}$$

Calculating the smallest difference between two current sources, one obtains

$$min\{\Delta I\} = \begin{cases} \frac{(x-1)^2}{x^N} \cdot I_{max} & \text{if } x < \frac{1 \pm \sqrt{5}}{2} \\ 0 & \text{if } x = \frac{1 \pm \sqrt{5}}{2} \\ \frac{2-x}{x^N} \cdot I_{max} & \text{if } x > \frac{1 \pm \sqrt{5}}{2} \end{cases} \qquad \text{(A.7)}$$

Appendix B

Node stability

There are two important nodes in the switched capacitor oscillator: V_+ and V_-. If one subtracts V_- from V_+, then V_3 is found. See Figure B.1.

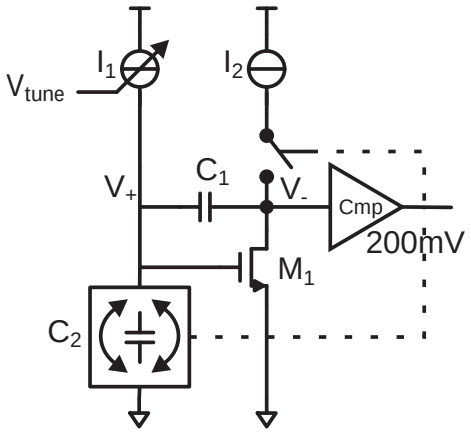


Figure B.1: Switched capacitor relaxation oscillator

If, due to some error (for instance charge injection of a switch or reconnecting a sample capacitor), a little bit of charge is injected into the oscillator at node V_+. The voltage at this node will rise a little bit according to $V = Q/C_2$. V_+ increases and so will V_{GS,M_1} and its drain current I_{D,M_1}.

As such the amount of extra charge will flow through C_1 to M_1 and will be integrated in C_1. This means that one needs to be careful with charge injection at this node as injected charge will be integrated then.

It is different for node V_-. A bit of extra charge cannot go anywhere as $I_{D,M_1} = I_1$ if C_2 is charged. This will raise the voltage at V_-. Node V_+ must follow, because, with respect to this extra injected charge, V_3 remains constant. The charge will then go to node V_+ through C_1.

The current flowing from V_- to V_+ will be integrated by C_1. But as V_+ increases so will V_{GS,M_1} and the drain current I_{D,M_1}. The extra charge, which just flew from V_- to V_+, will now go back from V_+ to V_- leaving V_3 as it was

before.

This means that node V_- can be regarded as being more stable, with respect to charge injection, than node V_+.