2008

# A jittered-sampling correction technique for ADCs

Jamiil A. Tourabaly
*Edith Cowan University*

# Edith Cowan University

# Copyright Warning

# A JITTERED-SAMPLING CORRECTION TECHNIQUE FOR ADCS

BY

JAMIIL A. TOURABALY

*A research thesis submitted for the award of*

**Master of Engineering Science**

*at*

School of Engineering and Mathematics

Edith Cowan University

Supervisor: Assoc. Prof. Adam Osseiran

December 2007

# USE OF THESIS

The Use of Thesis statement is not included in this version of the thesis.

**DECLARATION**

---

*I certify that this thesis does not, to the best of my knowledge and belief:*
*(i) incorporate without acknowledgement any material previously submitted for a degree or diploma in any institution of higher education;*
*(ii) contain any material previously published or written by another person except where due reference is made in the text; or*
*(iii) contain any defamatory material.*
*I also grant permission for the Library at Edith Cowan University to make duplicate copies of my thesis as required.*

Signature……. Date……

---

## Publications

Jamiil Tourabaly and Adam Osseiran, “A Jittered-Sampling Correction Technique for ADCs,” *Proceedings of the 4th IEEE International Symposium on Electronic Design, Test and Applications*, January 2008.

## Acknowledgements

Several people deserve recognition for their help for the accomplishment of this work.

First and foremost, I am very indebted to Associate Professor Adam Osseiran and Dr Stefan Lachowicz for their supervision and guidance.

A much deserve thanks must go to Alistair Kitchen for lending me his books and for his readiness to help at anytime.

A very special thanks to my wife, Tashreen, for making me laugh, supporting me and comforting me in ups and downs throughout all this.

To Dr Muhsen Aljada and Kelly Eng for being wonderful neighbours and for helping me in other adversities associated with life.

Finally, I wish to thank my parents for their support and trust.

# Abstract

In *Analogue to Digital Converters* (ADCs) jittered sampling raises the noise floor; this leads to a decrease in its *Signal to Noise ratio* (SNR) and its *effective number of bits* (ENOB). This research studies a technique that compensate for the effects of sampling with a jittered clock. A thorough understanding of sampling in various data converters is complied.

A novel design technique based on linear approximation is proposed to counter the effects of clock jitter in ADCs. The system consists of a circuit that performs linear approximation of the incoming signal to an ADC at time a possibly jittered clock is ticked to estimate the correct value of the sample. Since jitter is essentially caused by phase noise, the jitter is itself estimated using phase demodulation. To avoid introduction of even more noise sources passive and differential approaches have been selected.

Simulation results showed that the linear approximation technique used to estimate signal value at sampling instant does not cause significant distortions to the sample when compared to the error induced by jitter. With this approach, a 1.2 GHz sinusoid is sampled at a rate of 2.5GHz with various jitter levels. An improvement in the SNR of 8.09 dB was achieved. This corresponds to 1.34 bits of resolution gain in ENOB.

# Contents

Publications ..... iv
Acknowledgements ..... v
List of Figures ..... ix
1. Introduction ..... 1
1.1 Research Aims ..... 2
1.2 Thesis Organisation ..... 2
2 Background Review of ADC ..... 3
2.1 Figure of merits ..... 3
2.1.1 Offset and Gain Errors ..... 3
2.1.2 Nonlinearity Errors ..... 4
2.1.3 Effective Number of Bits ..... 6
2.1.4 Noise and distortion indicators ..... 6
2.2 ADC Architectures ..... 8
2.2.1 Flash (Parallel) Converters ..... 8
2.2.2 Pipeline A/D converters ..... 12
2.2.3 Successive Approximation Register ..... 14
2.2.4 Folding-Interpolating ..... 15
2.2.5 Sigma Delta ADC ..... 18
2.3 ADC Historical Background ..... 19
2.3.1 The Early Years ..... 19
2.3.2 Monolithic era ..... 21
2.3.2 High-Speed Converters ..... 22
2.3.3 Advancement in ADC Performance ..... 25
2.4 Jitter Analysis ..... 27
2.4.1 Noise Sources and Upconversion ..... 28
2.4.2 Noise up-conversion ..... 29
2.4.3 Jitter Effects on Sampling Systems ..... 30
2.4.4 Jitter Correction ..... 33
2.5 Conclusion ..... 33
3. Proposed Architecture ..... 34
3.1 Jitter estimation ..... 34
3.3.1 Phase Demodulation ..... 34

3.2 Approximation ..... 35
3.3 Simulation Results ..... 39
4. CMOS Implementation ..... 41
4.1 Differentiator ..... 41
4.1.1 Testing the Differentiator ..... 42
4.2 Mixer Design ..... 44
4.3 Oscillator Design ..... 48
4.4 Phase Demodulation ..... 53
4.5 Recommendations for a Large Bandwidth Amplifier ..... 55
4.5.1 Fully-Differential Amplifier Circuits ..... 58
5. Simulation analysis and Results ..... 60
5.1 SNR Improvement ..... 63
5.2 Limitations of Research ..... 66
6 Conclusion and Future work ..... 68
6.1 Future work ..... 68
Appendix A ..... 70
Appendix B ..... 72
References ..... 76

# List of Figures

Figure 2.1: Offset error in an ADC....4
Figure 2.2: Non-monotonic ADC with missing code....5
Figure 2.3: FFT of an output sequence of an ADC....7
Figure 2.4: Simplified Flash ADC Architecture [6]. ....9
Figure 2.5: Differential Amplifier....10
Figure 2.6: Preamplifier array with resistive averaging network [8]....11
Figure 2.7: Comparator offset cancellation techniques. (a) input offset storage and (b) output offset storage....12
Figure 2.8: Pipeline ADC architecture....12
Figure 2.9: Successive Approximation Register topology. ....14
Figure 2.10: 16-bit weighted capacitor DAC....15
Figure 2. 11: Folding-interpolating ADC. ....16
Figure 2.12: Oversampling in sigma delta converters ....18
Figure 2.13: Noise shaping in sigma delta converters. ....19
Figure 2.14: DAC 08 8-bit 85ns DAC, 1975 reproduced from [3]....21
Figure 2.15: Optical Sample and hold switch and dummy circuit for feedthrough cancellation. ....23
Figure 2.16: CMOS latch with positive feedback and overdrive recovery transistor..24
Figure 2.17: Time stretching in optical converters ....25
Figure 2.18: Historical trend in P [12]. ....26
Figure 2.19: Historical trend in F [12]. ....26
Figure 2.20: Historical trends in sampling speed and resolution [12]. ....27
Figure 2.21(a),(b): Spectrum of an ideal and actual oscillator. ....28
Fig 2.22: Four stage ring oscillator. ....30
Fig 2.23: Block diagram of feedback oscillator....30
fig 2.24: Performance limiters due to various noise and error sources. ....32
Figure 3.1: Graph of Jitter error and Taylor approximation error versus jitter....37
Figure 3.2: Graph of Jitter error and Taylor approximation error and linear Approximation error versus jitter over ½ cycle. ....37
Figure 3.3: Graph of Jitter error and Taylor approximation error and linear approximation on a logarithmic scale. ....38
Figure 3.4: Block diagram of the overall proposed system ....39

Figure 3.5: PSD of original, jittered, corrected and linear approximated sinusoid. ....40
Figure 4.1:Schematic of an active differentiator using fully differential amplifier. ....41
Figure 4.2: (a)Schematic of a passive differentiator and (b) in differential implementation. ....42
Figure 4.3(a): Frequency response of differentiator ....43
Figure 4.3(b): Phase response of differentiator....43
Figure 4.4(a): Sinusoidal excitation to differentiator....44
Figure 4.4(b): Transient sinusoidal response of differentiator....44
Figure 4.5: Potentiometric mixer ....45
Figure 4.6: Double Balanced Gilbert Cell Mixer....46
Figure 4.7: Double Balanced Passive Mixer....46
Figure 4.8: Mixer simplified equivalent circuit ....47
Figure 4.9: Transient response of mixer when excited with 2 sinusoids of 1GHz ....48
Figure 4.10: Transient response of mixer when excited with sinusoids of 1GHz and 27MHz ....48
Figure 4.11: Schematic of the VCO....49
Figure 4.12: (a) Test circuit to determine the negative resistance. (b) Equivalent half circuit of (a)....50
Figure 4.13: Transient simulation of oscillator....52
Figure 4.14: Oscillation spectrum. ....52
Figure 4.15: Perturbed oscillation spectrum ....53
Figure 4.16: Spectrum of the input signal to the LPF. ....54
Figure 4.17:Single stage of a CML buffer. ....56
Figure 4.18:Common Drain(source follower) stage. ....58
Figure 4.19:Fixed gain closed loop fully differential amplifier. ....59
Figure 5.1: Schematic of interconnected system ....60
Figure 5.2: Graph of delay correction to input signal versus improvement in SNR for a jitter of 5.1pS. ....61
Figure 5.3: Graph of autocorrelation of required correction signal and cross-correlation with actual correction signal from output of mixer. ....62
Figure 5.4: output of mixer and envelope detector. ....63
Figure 5.5: Spectrum of corrected and noisy signals ....64
Figure 5.7: Two tone test showing intermodulation products. ....65
Figure 5.8: Improvement in SNR versus Jitter. ....66

Figure B.1: Oscillator with signal conditioning circuit. .....72
Figure B.2: Phase demodulator.....73
Figure B.3: Mixer.....74

# 1. Introduction

Today's digital processing reliability has pressured evolution of data converters to extreme performances. In this line, much interest was expressed to bring the antenna closer to the processor in 'future proof' Software Radios (SWR). Even from this simplistic description, we can readily deduce a stringent requirement on the data converter within the design. For example, to maintain 'future proofness' and to keep-up with GSM and IMT-2000 standards, 14 bits resolution [1] and sampling above Nyquist rate is required. A sampling frequency of 2.5 times the Nyquist rate is a good engineering practice in this case. While sampling speeds of *Analog to Digital Converters* (ADCs) met the demand, general decline in resolution is observed in commercially available ADCs as sampling speed increases.

Recently, Walden [2] noted a fall of ~1bit for every doubling of the sampling speed. The fall off in resolution is attributed to limitations associated to different noise sources and distortions that worsen on increase of sampling rate and/or input frequency. As shown in more detail in the section 2.4.3, jitter in the sampling instant is the major limitation concern for the GHz to tens of GHz region. A reported technique to reduce jitter is to bandpass filter the incoming clock signal. However, the spectrum of a jittered clock shows a $1/f$ decline with frequency offset. This means that even high order filtering will only trim down the jitter to a limited extent, since most of the energy of the jitter noise is concentrated very close to the oscillation frequency.

The main source of jitter in oscillators is device flicker noise. Active devices exhibit flicker noise as a result of crystal imperfection that traps and releases charges in a random fashion. In an oscillator loop this is converted into amplitude and phase noise or a combination of both, depending on his occurrence in the oscillator cycle Hajimiri [1999]. Amplitude noise is easily removed from the oscillator loop by a limiter or inherently clamped by factors including limited supply rails voltage. Phase fluctuation, on the other hand persists indefinitely.

In this study it is important to differentiate between clock skew and clock jitter. Clock skew is a deterministic drift of a clock signal that is fixed or oscillatory in nature. Such phenomenon causes a decrease in the *Spurious Free Dynamic Range* (SFDR) of an ADC. Clock jitter, on the other hand, is random in nature and results in a rise in the noise floor of the signal spectrum and consequently decreases the *Signal to Noise ratio* (SNR).

Because of its complexity, jitter correction has received little attention. Existing jitter correction is in part achieved by efforts to improve the crystal structure of the die during fabrication, thereby reducing the source of jitter.

In multi channel converters, cyclic clock skew is present as a result of clock path mismatch. Compensation was reported to have been used to correct this form of timing error. However such technique has not been found in literature for correcting jitter Iroaga [2005].

## 1.1 Research Aims

The goal of this study is to design and simulate a novel compensation technique based on linear approximation to correct the adverse effect of sampling clock jitter, using commercially available SPICE-like software. The experiment will be set-up to investigate the resulting improvement in different metrics of ADC performance, eg. SNR, SFDR.

## 1.2 Thesis Organisation

Chapter 2 introduces a review of ADC architectures and historical trends. Sources of jitter and its effects are also presented along with attempts to tackle the degradation of SNR associated to jittered sampling. Chapter 3 develops the theory behind the proposed architecture and expected results. Chapter 4 describes the CMOS implementation of the building blocks of the scheme. Chapter 5 describes the interconnections of the building blocks, the test environment and setup. Chapter 6 Concludes the thesis and discusses future work.

# Chapter 2

## 2 Background Review of ADC

This chapter presents the various figure of merits associated with ADCs. A historical review of ADCs and trends are illustrated. Jitter noise and it sources are also introduced.

### 2.1 Figure of merits

A set of terms is commonly used in literature to quantitatively describe and compare performances of ADCs and DACs. Performance indicators can be determine quasi-statically or dynamically [2]. Quasi-static errors occur on slowly varying signals. Dynamic errors are additional errors that occur on faster time varying signals. The offset error, gain error and non linearity measurements that include differential nonlinearity (DNL) and integral nonlinearity (INL) are quasi-static, while dynamic measurement includes signal to noise ratio (SNR), spurious free dynamic range (SFDR) and two-tone intermodulation distortion. Dynamic analysis is performed in frequency domain in the form of a Fast Fourier Transform (FFT) on a sequence of output samples of the ADC under test.

#### 2.1.1 Offset and Gain Errors

Ideally the transfer characteristic of an ADC is a straight line. However, this would signify the requirement for an infinite number of quantisation levels. Accordingly in practice the transfer characteristic is implemented as a stair case. With each stride representing a quantisation level and its mid point crossing a straight line. The deviation of this line from the ideal one represents the offset of the comparator as shown in figure 2.1. This error affects all points by the same amount and is usually compensated by trimming. It is interesting to note that the offset error also causes a zero error.

A gain error is represented by a difference in slope of the actual and ideal transfer functions. Gain error is calculated after offset compensation and can be mathematically scribed as:

$$E_{gain} = \left( \frac{V_{1...1}}{V_{LSB}} - \frac{V_{0...01}}{V_{LSB}} \right) - (2^N - 2) \tag{2.1}$$

Similar to offset error, gain error affects all points by same percentage and can therefore be compensated through trimming.

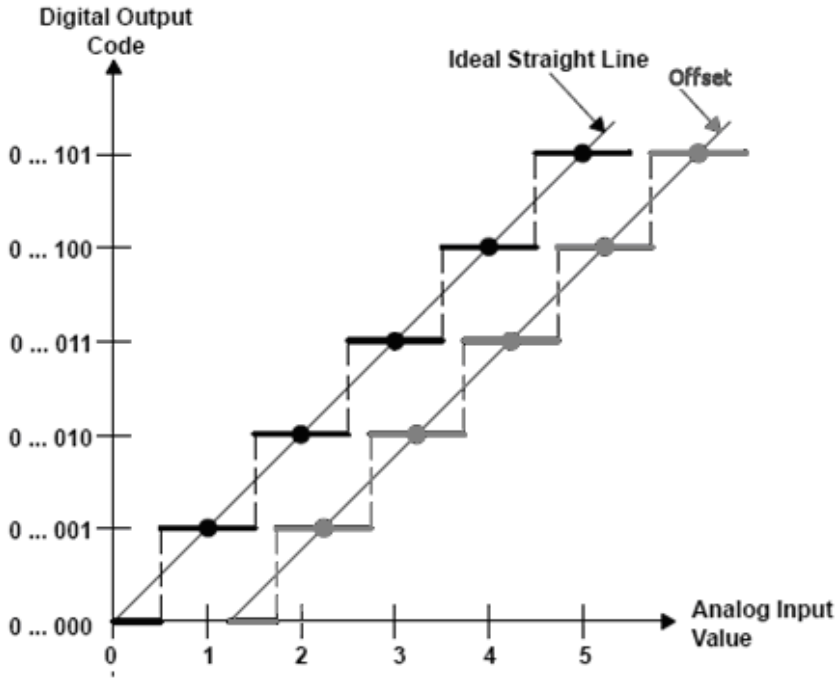



*Figure 2.1: Offset error in an ADC*

### 2.1.2 Nonlinearity Errors

Each step width for an ADC (or step height for a DAC) in its transfer characteristic is ideally 1 LSB. The DNL is the difference of the actual step width and 1 LSB. Ideal converters have a DNL of 0, while a maximum DNL of 0.5 represents step width varying between 0.5 LSB and 1.5 LSB [3]. A maximum DNL of less than ±1 LSB is a sufficient condition for monotonicity of a data converter. A monotonic ADC will output increasing digital codes for increasing analogue input and vice-versa for DAC. This can be seen in figure 2.2. Non-monotonic converters, when implemented in a closed loop system, can result in unpredictable system failure by inverting the intended feedback sign.

While sweeping the input voltage of an ADC through out its range, all digital output codes are expected to appear one after the other, in practice however, this is not always the case. Figure 2.2 shows the transfer characteristics of an ADC that has a DNL of -1LSB at code transition 101 to 110; the result is that 110 will never appear at the output. Such an ADC is said to have *missing codes*. DAC does not suffer from missing codes. For each digital code at the input, a corresponding analogue value is expected.

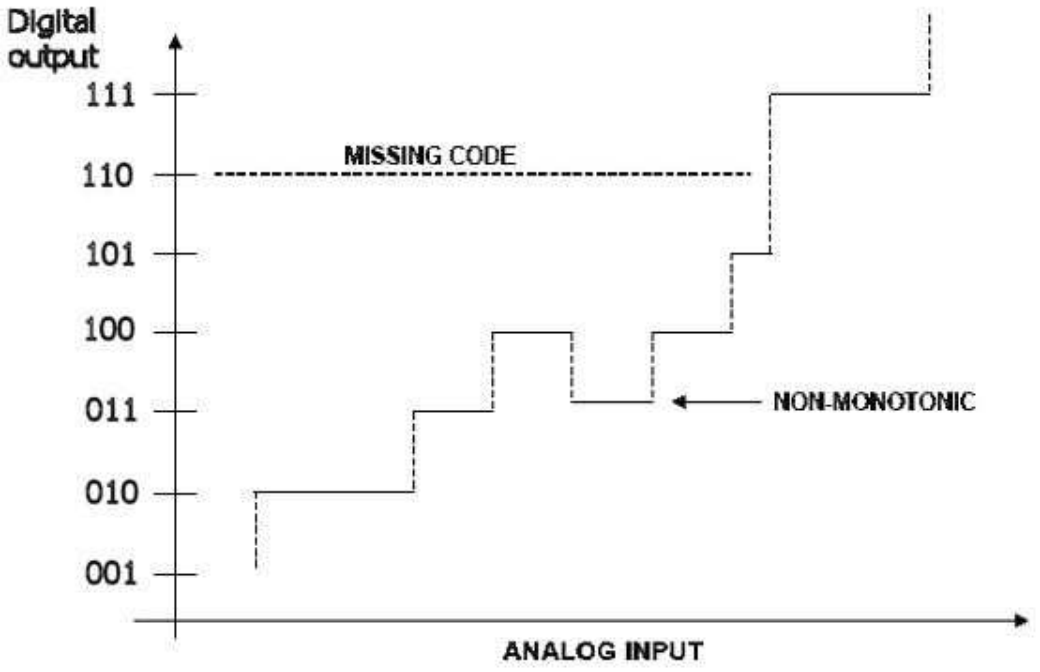



*Figure 2.2: Non-monotonic ADC with missing code*

INL is the integral of DNL errors and assess how much the transfer characteristics deviates from a straight line. The straight line normally used is a best fit or one through the endpoints. A maximum INL of less that 0.5LSB is another sufficient condition for monotonicity. The INL also gives an indication on the accuracy of the converter. For instance, a 15-bits ADC with a maximum INL of 4LSB gives an accuracy of $\frac{4}{2^{15}} = \frac{1}{2^{13}}$ which is equivalent to an accuracy of 13 bits having no missing codes.

Neither DNL nor INL errors can be easily compensated for.

### 2.1.3 Effective Number of Bits

The degradation of accuracy, with respect to the stated number of bits, calls for another figure of merit that assesses the actual accuracy of the converter. The Effective Number of Bits (ENOB) expresses the accuracy to the overall noise and distortions. An ENOB of *n*-bit corresponds to the noise and distortion of an ideal *n*-bits ADC [4]. An ideal ADC in this definition is one which suffers from quantisation noise only.

To gauge the overall performance in terms of sampling rate, ENOB and power dissipation all together, two widely used figures of merit F and P are used.

$$P = 2^B \bullet f_s \tag{2.2}$$

$$F = \frac{2^B \bullet f_s}{P_{diss}} \tag{2.3}$$

Here, B is the ENOB, $f_s$ is the sampling frequency, and $P_{diss}$ is the power dissipation. While P evaluates the combined performance of resolution and speed, F evaluates the power efficiency with respect to the resolution and speed.

### 2.1.4 Noise and distortion indicators

To analyse the dynamic performance of an ADC a pure sinusoid is applied to its input and a FFT of the output is observed. Such a typical output is shown in figure 2.3. The peak at 10 KHz represents the applied signal; all other spectral lines represent noise and artefacts of the conversion. These include quantisation and thermal noise, comparator ambiguity and jitter in the arrival of the sampling clock.

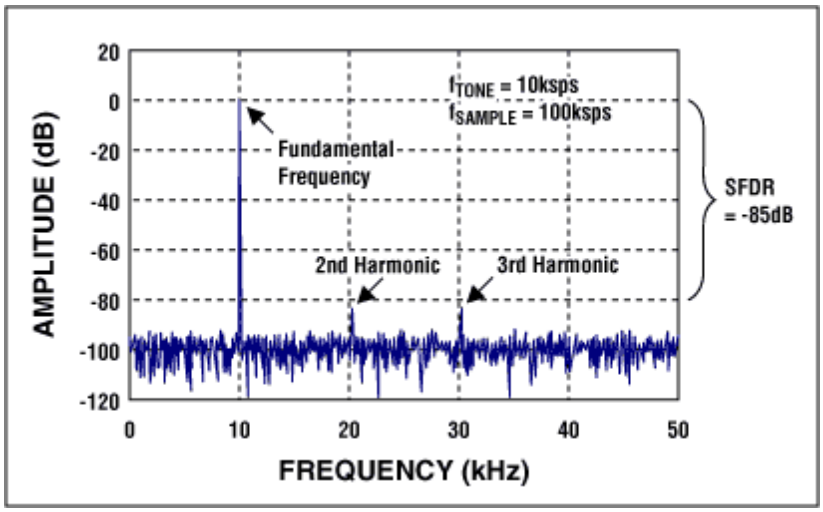



*Figure 2.3: FFT of an output sequence of an ADC.*

The Signal to Noise Ratio (SNR) is the ratio of the signal power to the noise. However, the definition of the noise in this calculation differs from manufacturer to manufacturer [3]. The most common definition is to include all noise and exclude the harmonics [4, 5]. Harmonics are typically considered up to the $10^{th}$ occurrence. The total spectral contribution of each harmonics is termed the Total Harmonic Distortion (THD). THD is calculated as the rms value of the harmonic components of the output signal, excluding the fundamental, and expressed as a percentage of the rms value of the fundamental. Dynamic nonlinearities cause THD to increase on increasing input frequency. Thus, change of THD versus frequency gives a good assessment of dynamic nonlinearities. SNR and THD; combined give the Signal to Noise and Distortion Ratio (SINAD or SNDR).

Still applying a pure sinusoid, the Spurious Free Dynamic Range (SFDR) is the ratio of the signal amplitude to the strongest non signal component within the spectrum of interest. SFDR, like THD and SNR, is typically expressed in dB. Large spur that can be seen on a FFT might not necessarily affect SNR by much, but will significantly reduce SFDR. Spurs prevent ADC from processing small signals, because it will be buried in much larger distortion. This has adverse effects on the input dynamic range. Characterisation of ADCs typically involves a plot of SNR and SFDR versus input frequency.

In most applications, an ADC is expected to faithfully convert any signal having spectral components below the Nyquist frequency, $f_s/2$. But many ADCs suffer from severe SNR degradation well below that rate. For this reason, the Effective Resolution Bandwidth (ERBW) is normally used in the place of the sampling frequency in equations (2.2 and 2.3). ERBW is defined as the frequency $f_{ERBW}$ at which the SNR falls by 3 dB with respect to the low frequency value.

## 2.2 ADC Architectures

### 2.2.1 Flash (Parallel) Converters

Perhaps the most straight forward approach to create an N-bits ADC is to have individual references for each of the $2^N$ output codes. The input signal is fed to $2^N$-1 comparators that have individual references obtained from a resistor string as shown in figure 2.4. While the parallel architecture makes flash converters one of the fastest, its exponential $2^N$ complexity relationship to the resolution renders it one of the less accurate. Each comparator having a reference which is smaller than the input voltage outputs logic '1', whilst the others outputs logic '0'. This in effect produces a thermometer code. A decoder, usually implemented as a ROM, is used to convert the thermometer code to binary.

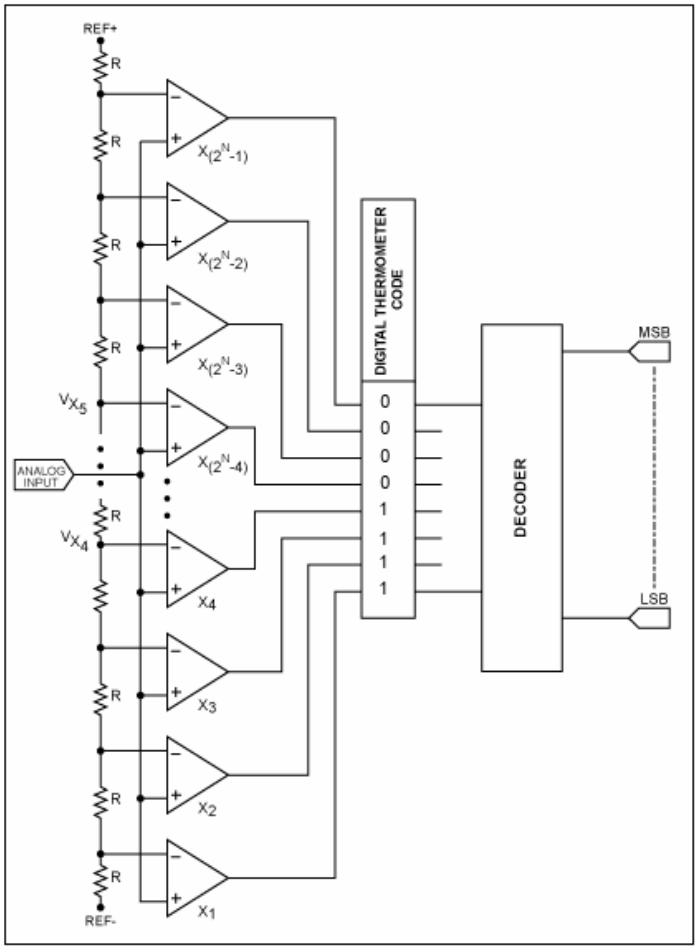



*Figure 2.4: Simplified Flash ADC Architecture [6].*

Flash converters suffer from bubble errors (also called sparkle codes). A valid thermometer code consists of a series of all ‘0’ and all ‘1’, like 00001111. An out of place ‘0’ or ‘1’ is called a bubble and is the result of comparator ambiguity, imperfect input settling time, or comparator timing mismatch [6]. Most modern flash converters use encoding techniques that minimises the effect of bubble errors.

Due to transistor mismatch, the comparators do not settle at the same instant. This adversely affects the overall performance of the ADC, if the input signal is changing before all comparators are settled. The most serious impact is a drop-off in the SFDR as the input frequency increases. A track and hold (T/H) circuit is added to the input line to assess this setback. Non negligible input currents to the comparators cause errors in the voltages of the nodes of the resistor string. This is often called resistor bowing and severely increases the INL. However the deviation is greatest at the centre

node of the resistor string and it is usually solved by adding extra circuitry to force the centre node to the correct voltage.

Component mismatch that is a consequence of non uniform doping of the different transistors during manufacture causes in gross calibration, DNL errors and even non-monotonicity. This can be corrected to some degree by using larger transistors, but this is area greedy and increases input parasitic capacitance. Attaining monotonicity while maintaining high resolution dictates that offsets need to be tightly controlled or compensated for. To reliably achieve monotonicity necessitates that [7]

$$\sigma_{offset} = \frac{1}{4}\text{LSB} \tag{2.4}$$

Where $\sigma_{offset}$ is the standard deviation of random comparator offset.

Achieving good linearity (i.e., DNL < 0.5) places more stringent requirements:

$$\sigma_{offset} = \frac{1}{8}\text{LSB} \tag{2.5}$$

Efficient techniques to alleviate those requirements include offset storage and resistive averaging. Comparators usually consist of a CMOS latch preceded by differential preamplifier. From figure 2.5 it follows that any mismatch in the 2 transistors will cause an offset in the amplifier and this error will be translated to the latch. Choi and Abidi [8] presented a spatial filtering technique consisting of a resistor network for offset averaging.

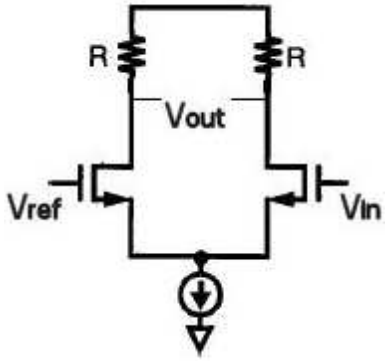



*Figure 2.5: Differential Amplifier*

Averaging resistors connects the adjacent output nodes and translational symmetry of the filter is maintained by a circular arrangement as shown in figure 2.6. In such arrangement, in order to maintain an undistorted impulse response at the boundaries of the full scale dummy preamplifiers are added and compare the input voltages with references beyond the actual full scale [8].

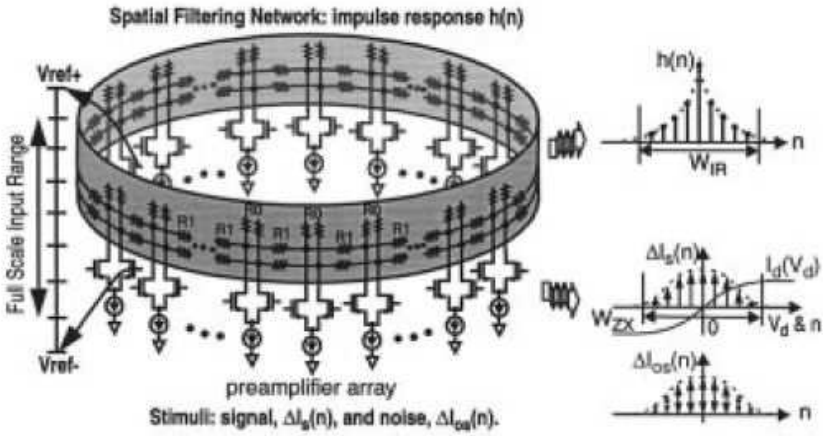



*Figure 2.6: Preamplifier array with resistive averaging network [8].*

Offset storage is another technique used to tackle component mismatch and the resulting comparator offset problem. This scheme consists of a switched-capacitor circuit that would store the offset during tracking phase and compensate for the stored offset during the conversion phase. Offset storage subsists as two topologies, namely *Input Offset Storage* (IOS) and *Output Offset Storage* (OOS). Each of these consists of a preamplifier, offset storage capacitors and a latch. IOS involve closing a unity gain loop around the preamplifier and storing the offset on input coupling capacitors. With OOS, the inputs of the preamplifier are shorted and the offset at the output are stored on coupling capacitors. Figure 2.7(a) and (b) illustrate these two approaches.

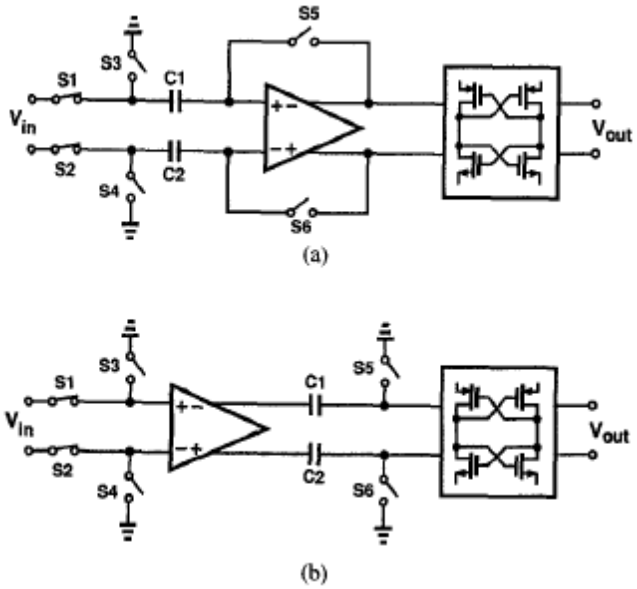



*Figure 2.7: Comparator offset cancellation techniques. (a) input offset storage and (b) output offset storage*

### 2.2.2 Pipeline A/D converters

Perhaps at the expense of being slightly slower, most of the disadvantages of flash converters are tackled by the pipeline topology. Specifically they require less area, dissipate less power, have less input capacitance, and the offset requirements are less stringent that with flash counterparts. This makes pipeline architecture currently most popular approach for medium-accuracy high-speed ADC.

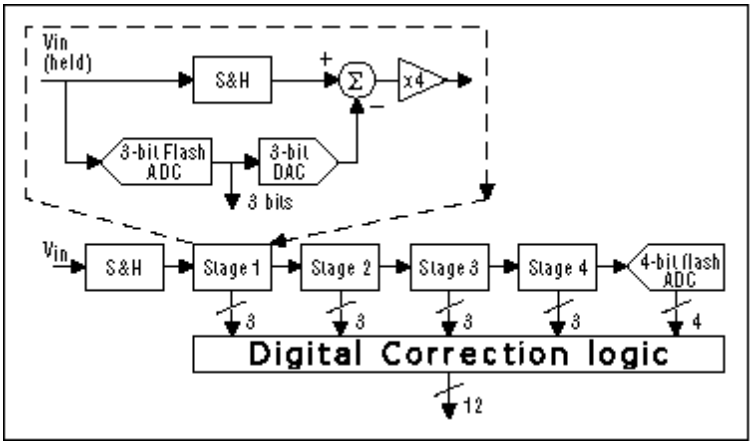



*Figure 2.8: Pipeline ADC architecture.*

Pipeline converters consist of several consecutive stages, each containing a SHA, low resolution ADC and DAC and a summer with a fixed gain amplifier. The principle of

operation relies on the first stage to make a coarse conversion, and pass the residual quantization error to the next stage for a relatively finer conversion and in turn pass the residue to the next section. The fixed gain amplifier reduces the requirements of the next stage [9]. It also simplifies implementation by standardising the stages. The last stage is freed from the need to calculate the residual quantisation error which results in a basic flash implementation.

The overall resolution is less that the total individual contribution from each stage, as $x(t)$ shown in figure 2.8. This should be met for correcting overlapping errors [10]. Considering the case of an 8-bit ADC implemented in 2 stages[1] with the first one 4-bit accurate, the following holds for an ideal 8-bit ADC,

$$V_{ref} \bullet \sum_{n=0}^{7} D_n \bullet 2^n = V_{in} + V_q, \quad -\frac{1}{2}V_{LSB} < V_q < \frac{1}{2}V_{LSB} \tag{2.6}$$

where $D_n$ is the n$^{th}$ digital output starting with 0 for LSB. However for a non ideal case with an absolute accuracy of 0.5LSB, the quantisation error, $V_q$ would be of the range $-V_{LSB} < V_q < V_{LSB}$. Keeping the same scale, we have for $-8V_{LSB} < V_q < 8V_{LSB}$ an ideal 4-bit ADC. Thus for a real ADC, $V_q$ is now bonded within 32LSB, and requires a 5-bit resolution for the last stage [11].

The chain architecture of the pipeline converter makes straightforward implementation too slow in a high resolution ADC, since the last bit would have to wait for the quantisation error to ripple through each of the stages of the entire converter. To increase throughput in this case, memory elements in the form of digital delays and shift registers are added to the design so that once the first stage has settled, it does not sit idle while the remanning are being found, instead it starts working on the next sample.

1. Pipeline ADC are also called sub-ranging ADC and the special case having only 2 stages is also known as half-flash or two step flash.

### 2.2.3 Successive Approximation Register

Successive Approximation Register (SAR) is a cheaper and yet more efficient mid-range implementation for speed and resolution. SAR uses a recursive trial and error algorithm to test each bit value. A typical diagram is shown in figure 2.9. The use of only one comparator and its algorithmic nature allows low power implementations with mean of 10 mW/channel. This is reflected by an exceptional $F$-figure of $5.5x10^{12}$ achieved in 2006 [12]. SAR sample rates are inversely proportional to speed because the conversion requires one clock cycle to resolve each bit of resolution of the output. Commercially available SAR has stated number of bits (N) in the range of 7 to 22 and a sample rate of 10 kSps to 6.7MSps.

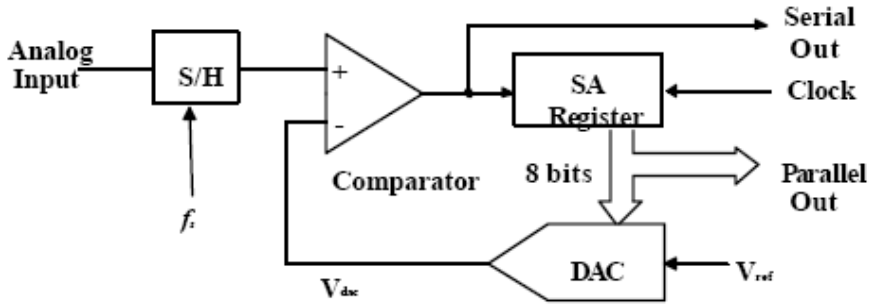



*Figure 2.9: Successive Approximation Register topology.*

SAR algorithm involves testing logic '1' in the MSB position with all other bits '0' and compares its value with the input with the help of a DAC and Comparator. The resulting signal is used to decide whether to keep that '1' in the MSB before proceeding to the next bit. The DAC and the comparator represent the most critical components in this architecture. The linearity of the DAC often determines the overall linearity of the ADC. A common, accurate and efficient implementation of the DAC in a SAR is the switched capacitor method [13]. Figure 2.10 depicts the basic architecture of a 16-bit weighted capacitive DAC implementation. This type of DAC has inherent sample and hold capability and frees the designer for the need of adding a SHA. The DAC consists of an array of N+1 capacitors for an N-bit resolution. The DAC operates successively in 3 modes. In the first step, which is the sampling mode, the inputs of the comparator as well as the common plates of the capacitors are shorted to the ground while the other capacitor plates are charged to $V_{in}$. After

acquisition the common plates are disconnected from the ground and instead the other terminals of the capacitors are grounded, effectively driving the common terminal to -$V_{in}$. Next, The MSB capacitor is connected to $V_{ref}$. This raises the common terminal to -$V_{in}$ + ½ $V_{ref}$. If the common terminal voltage remains negative, the comparator output will be a logic '1' which is used to keep the MSB capacitor clamped to $V_{ref}$. In case the common node voltage was positive the capacitor is reconnected to $V_{in}$. Same is repeated to the rest of the capacitances until the LSB capacitor is reached. The dummy capacitor is required for accurate division by 2 with each capacitor selected.

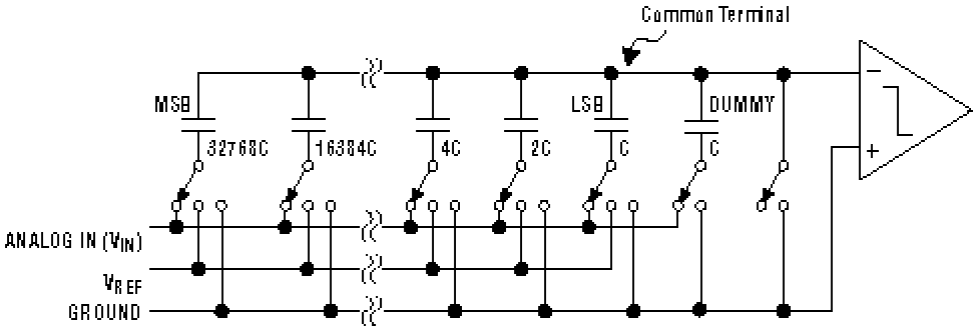



*Figure 2.10: 16-bit weighted capacitor DAC*

## 2.2.4 Folding-Interpolating

The $2^N$ comparators in flash topology causes high input capacitance and requires large area. Folding architecture is an attempt to solve this shortcoming with least compromise on sampling speed. The topology consists of a parallel implementation of folding amplifiers that feeds an interpolation string of resistors that will in turn trigger latches as shown in figure 2.11 [14]. The basic function of the folding amplifiers is to convert a linearly increasing (or decreasing) input signal into a clipped sinusoid-like signal. The number of transitions for a single folding amplifier while $V_{in}$ sweeps FSR is referred to as the *folding rate*.

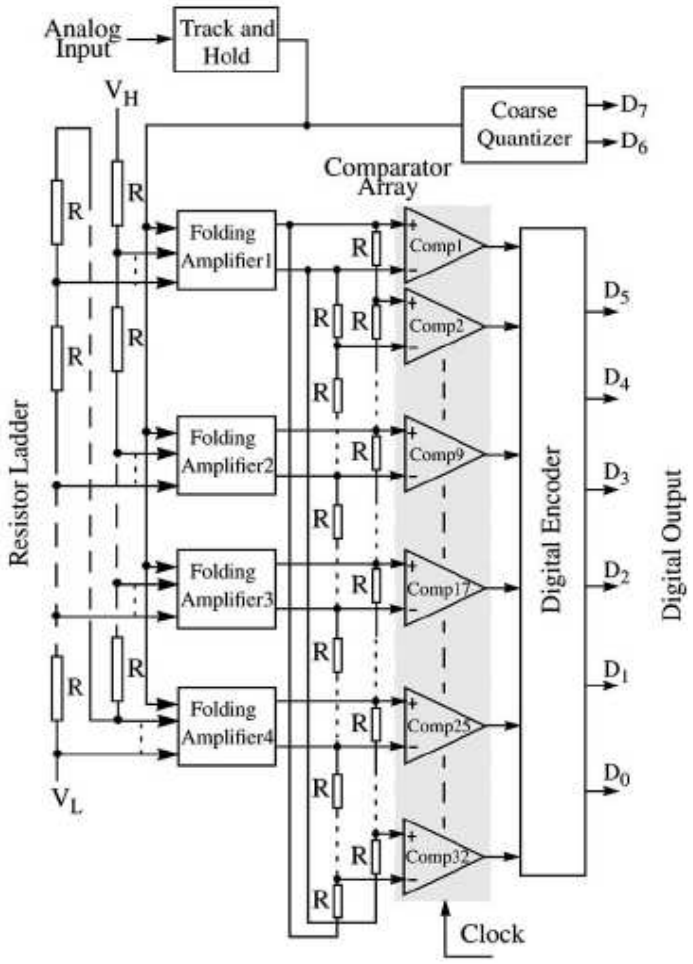



*Figure 2. 11: Folding-interpolating ADC.*

A folding amplifier having a certain folding rate requires an equal number of voltage references usually obtained from a preceding resistor string. The folding rate determines the number of bits required for the MSB converter given the overall resolution desired [15]. The coarse quantizer in the example shown is 2 bits wide and thus divide the FSR in to 4 parts; can $V_{in}$ be in the ranges $0-\frac{1}{4}, \frac{1}{4}-\frac{1}{2}, \frac{1}{2}-\frac{3}{4}$, and $\frac{3}{4}-1$ wrt the reference. The comparator output produces thermometer codes for each of the MSB segments. The thermometer code is reversed in the segments $\frac{1}{2}-\frac{3}{4}$ and $\frac{3}{4}-1$. For example, as $V_{in}$ rises from 0 to $\frac{1}{4}$ the thermometer code sweeps 0……0, 0……01, up to 1……1. However as $V_{in}$ rises above $\frac{1}{4}$ to $\frac{1}{2}$ the thermometer code decreases back to 0……0. It is important note that the MSB

converter and the folding one works in parallel, saving it from localised idle time. Background literature [14] reports a folding–interpolating converter reaching 2 Gsamples/s with 8 bits resolution. Due to its low comparator count, folding ADCs have the lowest power to sampling speed ratio of $1.502 \times 10^{-6}$mW/sps [12].

A common issue in folding ADCs is the frequency multiplying of the input signal with the folding rate. This severely reduces the bandwidth of the converter. Fortunately, this problem can easily be solved by the introduction of a THA. Moreover, by keeping the signal constant over a period of time eliminates the necessity of synchronisation between coarse ADC block and the fine one. A trade off exists in implementing the THA as a front end versus a distributed topology after the first stage. The advantage of the former case is that the THA bandwidth is only required to match the input signal, whereas in the distributed case the signal bandwidth multiplied by the gain of the stage and average while in centralised topology the THA errors are amplified by subsequent stages [16].

Additional folds and zero-crossings after each stages increase accuracy and can be added with minimal hardware through current or voltage interpolation. Current interpolation involves the use of scaled current mirrors but proves to be power hungry and not very precise. On the other hand, voltage interpolation is implemented by the use of resistive strings that also provide offset averaging [16]. Choosing an overall folding and distributing it among folding stages sandwiched between interpolating and averaging circuits is subject to a number of tradeoffs. High folding rate in early stages decreases the amount of hardware required. But amplifiers with high folding rate have longer settling time and adversely affect the overall bandwidth of the converter. Besides, having high folding rate amplifiers in the final stages, where errors due to incomplete settling are less harmful, does increase the overall sampling rate but the area requirements are more intensive and the advantage over flash implementation decreases. For these reasons folding rate distribution should be tackled carefully.

While folding converters are the cheapest and most efficient in their sampling speed range, they remain ironically the least popular commercially.

### 2.2.5 Sigma Delta ADC

Plausibly one of the most challenging architecture to comprehend, sigma-delta converters are yet the most accurate ones. The only choice for the designer is sigma delta if the targeted resolution is more than 16 bits [12]. The analogue part of this converter is a 1 bit ADC and uses techniques like over sampling and noise shaping to increase resolution to an extent unmatched by others.

Oversampling principle is based on the fact that the noise energy is constant and independent on sampling speed. Thus sampling several times above Nyquist effectively spread the noise energy over a larger bandwidth and thereby lowering the noise floor. Low pass filtering and decimation is performed to filter off the noise outside the band of interest. This is illustrated in figure 2.12.

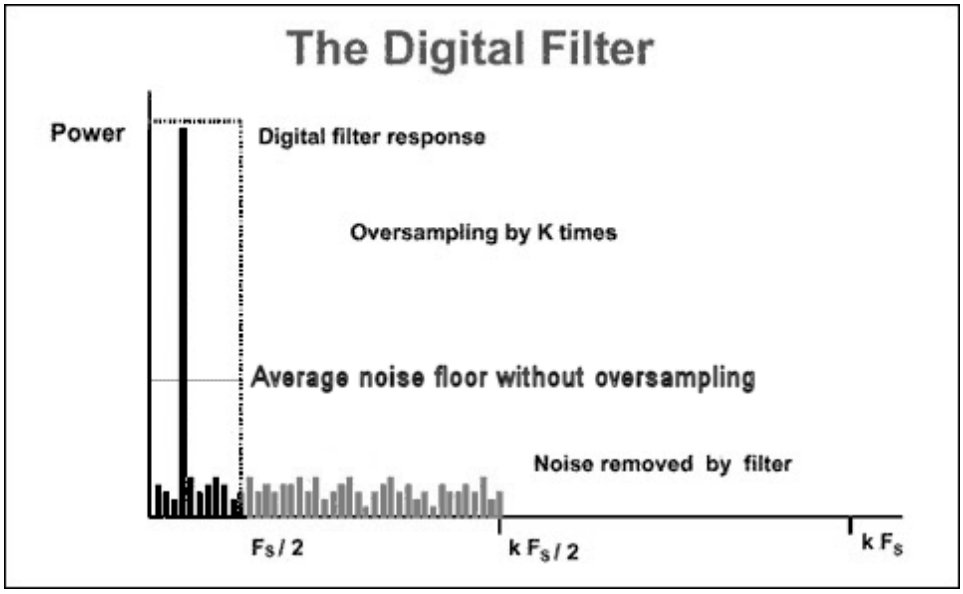



*Figure 2.12: Oversampling in sigma delta converters*

More is to be gained by performing noise shaping. This technique involves shifting the noise in the frequency domain outside the band of interest. Figure 2.13 shows a simple block diagram of a sigma delta converter equipped with an integrator that performs noise shaping. Since the density of '1's at the modulator output is proportional to the rate of change of the signal, there is a greater number of '1's for increasing input signal and vice versa for a decreasing input. By summing the error signal from the difference amplifier, the integrator is effectively a low pass filter to

the signal and a high pass filter to the quantization noise. This further separates the noise and a higher SNR is achieved.

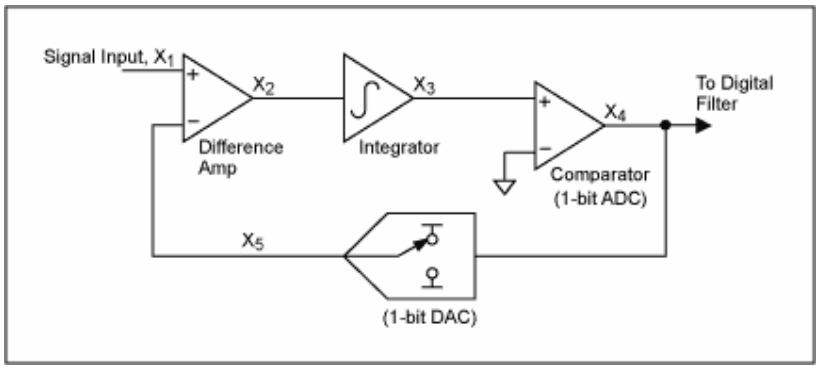



*Figure 2.13: Noise shaping in sigma delta converters.*

## 2.3 ADC Historical Background

### 2.3.1 The Early Years

The earliest form of data conversion was time division multiplexing (TDM) by M.B. Farmer in 1853 and J.M.E Baudot put it into practice by using a set of mechanical commutators as time division multiplexers in 1875 [11].

In a 1903 patent, Willard M. Miner reported results of experiments with the use of such kind of multiplexers. His findings loosely describe the minimum sampling frequency as "… a frequency, or rapidity approximating the frequency or average frequency of the finer or more complex vibrations which are characteristics of voice…". This statement was refined by Harry Nyquist in 1924. He modelled the telegraph signal as

$$s(t) = \sum_k a_k f(t - kT) \tag{2.7}$$

In the above equation $f(t)$ is the time domain representation of the basic pulse shape; $a_k$ is the amplitude of the $k^{th}$ pulse. Nyquist stated that if such a signal is passed through a channel of bandwidth W the pulse rate, 1/T, could not be increased beyond 2W. This is now known as the Nyquist criterion for sampling and is more commonly

written: If a B Hz band limited signal is sampled at regular intervals at a rate which is at least 2B Hz then the samples contains all the information in the original signal.

Prior to mid 50's, data converters was built on vacuum tube technology and was, thus, bulky and power greedy. Such converters had their main use in research and militarily for encryption. During mid 50's to early 60's Vacuum tube technology gradually faded out as they were being replaced by solid state electronics. In that era, the use of converters was still mainly military but development was much faster as compared to vacuum tube period. The table below summarises the major achievements in data converters in a time line.

TABLE 2.1: TIMELINE OF MAJOR DEVELOPMENT IN DATA CONVERTERS

| **Year** | **Achievement** |
|---|---|
| 1853 | Time Division Multiplexing |
| 1924 | Nyquist Criterion |
| 1937 | Pulse Code Modulation |
| 1939 | Reeve's Counting ADC |
| 1946 | Successive Approximation Register ADC |
| 1948 | Flash ADC with Electron Tube |
| 1949 | Grey Coding |
| 1950 | Delta Modulation, Differential PCM |
| 1952 | Voltage to Frequency Converter |
| 1954 | DATRAC: 11 bit, 50 Ksps vacuum tube ADC 500W power dissipation |
| 1956 | Subranging ADC |
| 1956 | Subranging with error correction |
| 1957 | Dual Slope ADC |
| 1962 | Sigma Delta Modulation |
| 1966 | HS-810: 8bit, 10Msps, 150 W |
| 1967 | Triple Slope ADC |
| 1969 | Successive Approximation Register realised with 14 7400 series IC |

| | |
|---|---|
| 1973 | Quad Slope ADC |
| 1978 | First Monolithic ADC: AD571 SAR, 10 bit, 25 μs. |
| 1988 | Bandpass Sigma Delta converter |

### 2.3.2 Monolithic era

Converters of the 70's were monolithic, modular and hybrid designs. Monolithic designs were slower and offered lower resolution than modular designs of that time. However, things changed in the 80's when bipolar technology became more reliable and gain popularity. DAC consisted of diffused resistor ladders and switches. Although monolithic in nature, early versions like the one shown in figure 2.14, required external components, voltage references and output buffers.

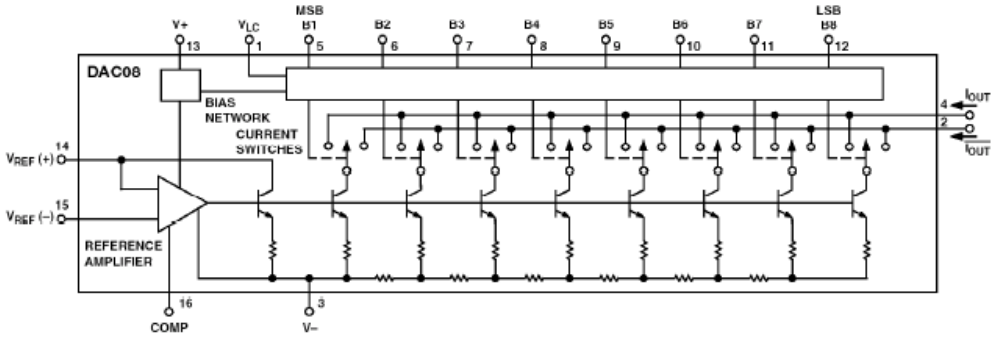



*Figure 2.14: DAC 08 8-bit 85ns DAC, 1975 reproduced from [3]*

The accuracy was also limited to 8 bits due to matching and tracking limitations of diffused resistors. Although DAC with laser trimmed thin film resistors came to existence, the resolution was still limited to 10 bits due to finite β gain of bipolar transistors. High power dissipation and the finite β gain impediment of bipolar circuits were solved by the introduction of CMOS circuits. *Sampling* monolithic ADCs appeared in the mid-80s. The added sample-and-hold, references and buffer amplifiers on-die circuits were simplified using BiCMOS technology.

### 2.3.2 High-Speed Converters

This section showcases ways by which data converter speeds are enhanced.

*Time interleaved converters*

Perhaps the most straight forward way to come back with the ever increasing demand for higher sampling speed is parallelism of an array of identical ADCs. This is performed in time-interleaved comparators; however the implementation does not consist of a mere parallel topology. Rather, an advance timing circuitry synchronises the different channels and recombination at output. As a rule of thumb, having N channels will increase the overall sampling speed up to N times the per-channel sampling speed. Thus, the complexity increase is proportional to the throughput increase [17]. This contrast flash implementation that is fast but its complexity increases exponentially with resolution. Instead slower but more accurate converters can be ganged together in a time interleaved fashion to gain speed. In $\Sigma-\Delta$ converters, the target sampling rate can be reached by using several parallel modulators, instead of faster and therefore costly fabrication process or using higher order modulators.

To benefit from the advantage of time interleaving in the tens of GHz input frequency range, optical sampling is preferred due to the lower jitter susceptibility required to achieve even modest resolution. Photonic sampling consists of optically triggering a photoconductive cell and holding the sampled value on a capacitor. However, photoconductive cell provide poor isolation between the input signal and the sampled value during 'off' mode. Nathawad *et al.* [18] report a photonic sample and hold switch comprised of low-temperature GaAs metal-semiconductor-metal photocell that provide an on-resistance $R_{on} < 100\Omega$ and off-resistance $R_{off} > 100M\Omega$. Nonetheless, it exhibits a relatively large feedthrough capacitance of 10fF. Albeit area inefficient Nathawad *et al* .showed that a dummy switch and differential amplification can get rid of this sample corruption as shown in fig 2.15.

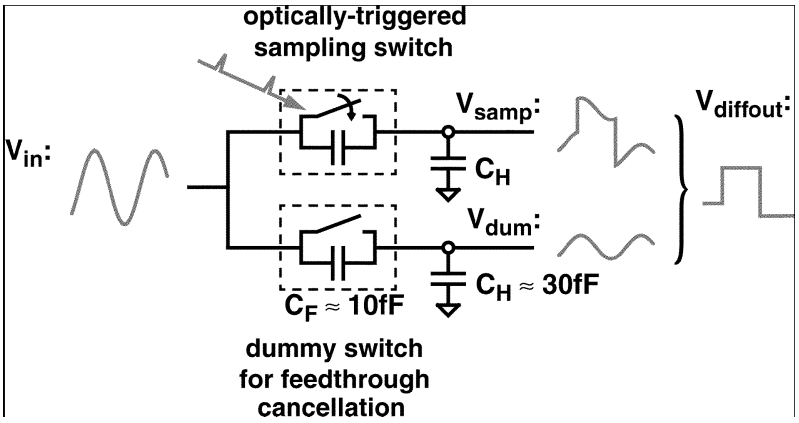



*Figure 2.15: Optical Sample and hold switch and dummy circuit for feedthrough cancellation.*

Gain and offset channel mismatches in time interleaved architectures cause undesired errors in the digital output of the converter. Channel to channel gain mismatch induces amplitude modulation of the signal while an offset mismatch introduces a phase error [19].

Furthermore, important design trade-offs exist in choosing a centralised THA circuit versus a distributed implementation in each channel. In the first technique, the THA handles the full bandwidth related to the sampling rate. The settling time of the THA is therefore the limiting factor of the sampling rate, how ever the benefits of time-interleaving is optimally achieved when the settling time of the comparators in each channel is the speed limiting factor. On the other had, distributed THA technique introduces a cyclic sampling jitter associated with clock path mismatches. This indeed adversely affects the SFDR of the converter.

*Comparators*

Since the speed limiting factor of ADC's is governed by the settling time of its comparators, it is only natural to try to optimize this to achieve faster sampling speeds. ADC comparators need to achieve both high bandwidth and gain. However, high gain amplifier have long settling time and there fore low bandwidth [20]. In most architecture, the voltage difference of $V_{LSB}/2$ to CMOS logic states and this is inherently requires a fairly high gain.

Fast ADC comparators achieve high gain and bandwidth by implementing the comparison circuit in a cascade of preamp, comparator, and CMOS latch. Each stage contributes a moderate gain and processes a high bandwidth, resulting in both a high overall gain and bandwidth.

To further reduce the settling time of the comparator, and enhance the voltage swing, CMOS latches has positive feedback [20-22], as shown in fig 2.16. This yield fast comparison, increases the voltage swing, and minimises chances of meta-stability and comparator ambiguity from occurring [22]. For fast overdrive recovery, an additional transistor is added to the latch and clamps the output to a tie during the precharge phase of the clock.

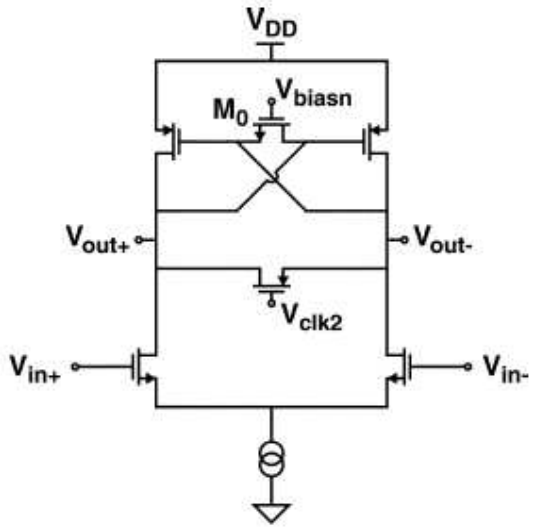



*Figure 2.16: CMOS latch with positive feedback and overdrive recovery transistor*

Further exploiting optical medium, fast converters in the range of 480 Gsamples/s has been reported using optical signal processing. In this scheme, the input signal is segmented and each segment is fed into a channel within a time-interleaved architecture. This differs from time interleaving used in used in electronics where only one sample is processed in a channel. Temporal segmentation is performed using a passive optical filter and eliminates the need of electronic demultiplexers [23]. Using optical dispersion techniques the signal is time stretched by a factor equivalent to the number of channels implemented. The sampler that follows the time stretcher effectively sees a slowed down version of each segment as shown in figure 2.17. This

decreases the requirements of the SHA as well as the quantizer. Recombination is performed in digital domain.

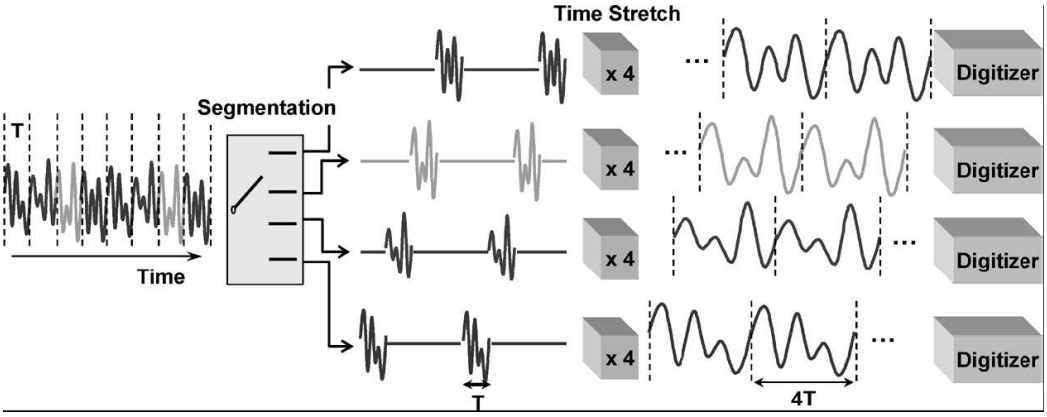



*Figure 2.17: Time stretching in optical converters*

### 2.3.3 Advancement in ADC Performance

Surveys [2, 12] shows performance evolution of ADCs over the past 20 years. Prior to 1997, year at which Walden's [2] data ends, $F$ have been increasing over time while, $P$ remained relatively constant. Walden attributed the improvement in $F$ as a result of more monolithic products and the stagnation of $P$ as a de-emphasis on research. However, subsequent breakthrough in both $P$ and $F$ was exponential as shown in figures 2.18 and 2.19. While sigma-delta converters do not always have the best $P$ they form the highest $F$ envelope in 1995-2004. The decrease in $F$ as from 2002 is attributed to the higher over-sampling rates required to overcome increased distortions in higher resolution sigma-delta converters of the period.

It is interesting to note that $F$ has a strong dependency on architecture while $P$ does not.

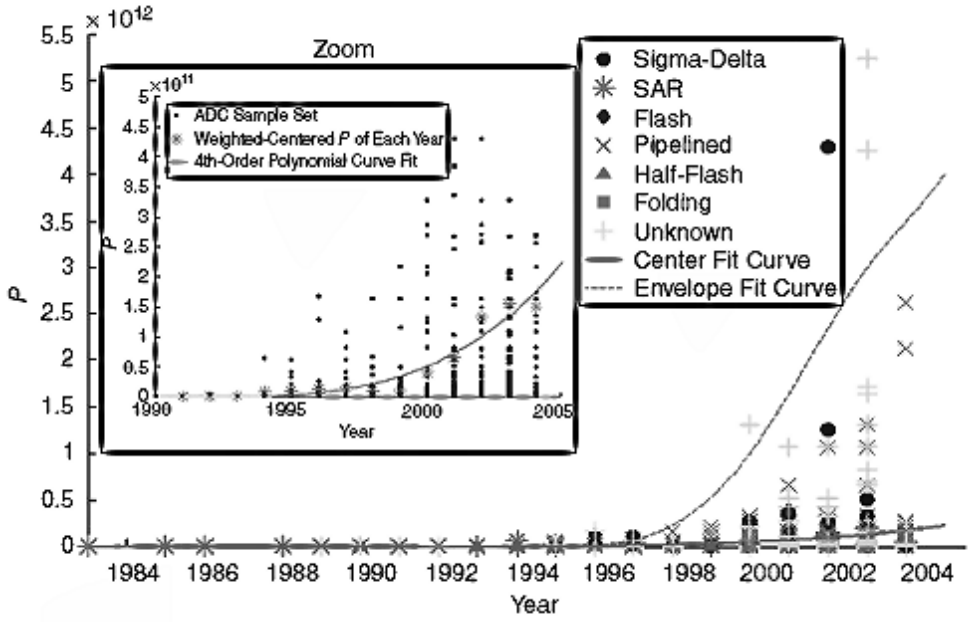



*Figure 2.18: Historical trend in P [12].*

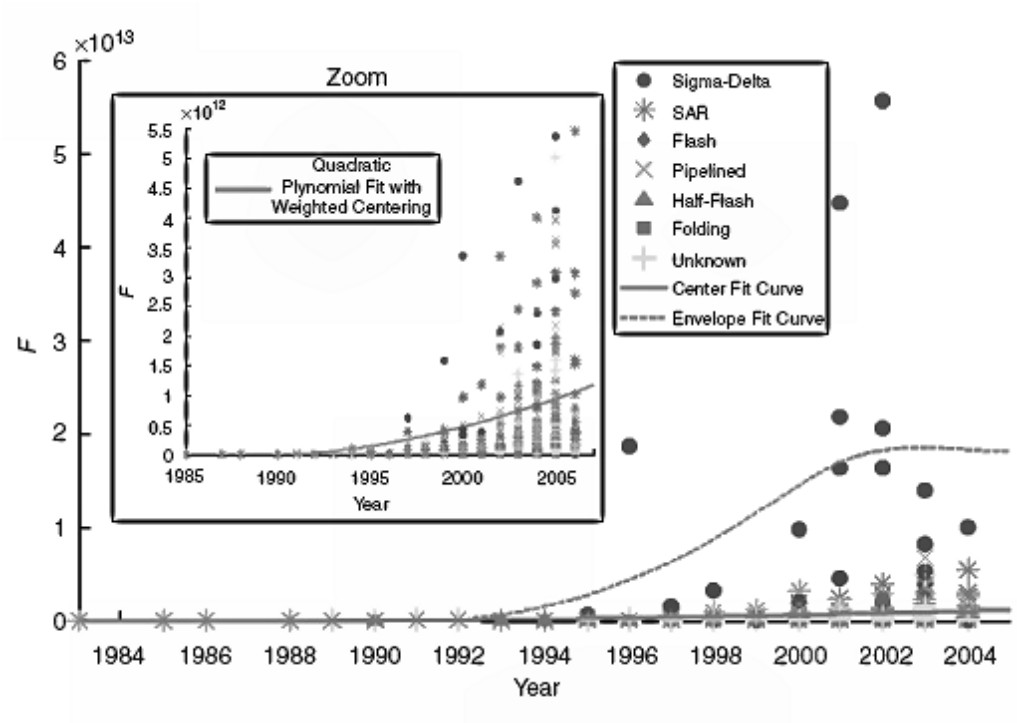



*Figure 2.19: Historical trend in F [12].*

Nascent applications have been the incentive for development of ADC technologies that resulted in the exponential $P$ increase. However, specific requirements for sampling speed and resolution have had different trends.

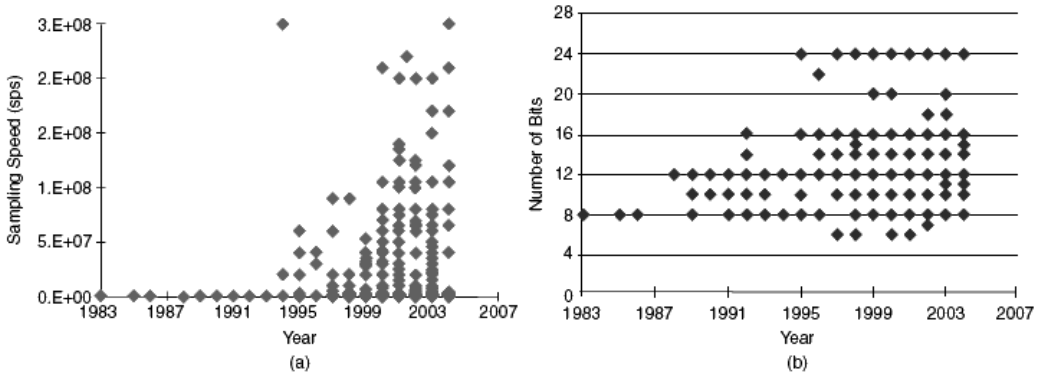



*Figure 2.20: Historical trends in sampling speed and resolution [12].*

Looking closely at figure 2.20, the demand for faster ADC kept on increasing while resolution envelope remained steady since 1995. This is explained by the fact that current applications such as 3G cellular and WiFi did not necessitate any greater resolutions. Radar systems, UWB and orthogonal frequency division multiplexing are expected to provide the push for even faster ADCs in the medium term. Power efficient devices that would fit portable devices are desirable in the short term.

## 2.4 Jitter Analysis

In an ideal oscillator transition spacings are constant, yet in practice transition spacings randomly vary due to noise inside the oscillator loop. Oscillators are subject to amplitude and angular noises. The former is easily removed by the use of a limiter, or inherently clamped by factors including limited supply rails voltage, whereas any fluctuation in the phase jitters the oscillation and persists indefinitely [24]. It is convenient to describe timing errors as 2 entities, namely clock skew and clock jitter [25]. Clock skew is a deterministic drift of a clock signal that is either fixed or oscillatory in nature and results in a loss of SFDR. Clock skew is often the result of non identical clock path in time interleaved converters. The randomness of the clock jitter on the other hand, raises the noise floor and consequently reduces the SNR of the sampled signal and affects both single and multi channel converters.

In time domain, jitter which is also referred to as phase noise, can be viewed as a sinusoid with perturbed zero crossings with a $\Delta t_i$ deviation from the ideal time instant. This can be translated in the phase deviation $\phi(\Delta t_i) = 2\pi f_c \Delta t_i$. The

mathematical definition of phase noise is the variance of $\phi(\Delta t_i)$, $E(\phi(\Delta t_i)^2)$. In frequency domain, this translates to sidebands that inversely decline with the offset from the frequency of oscillation. Figure 2.21 shows how this differs from the ideal line spectra expected from oscillators.

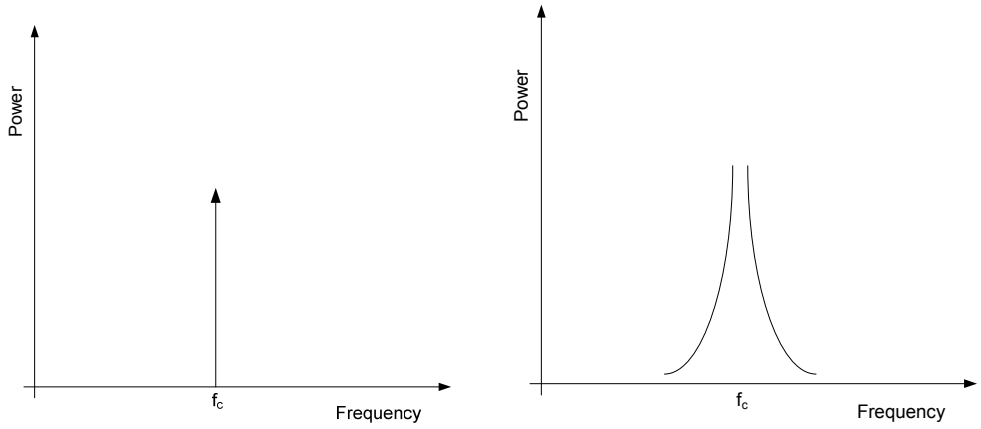



*Figure 2.21(a),(b): Spectrum of an ideal and actual oscillator.*

### 2.4.1 Noise Sources and Upconversion

Random movement of electrons in current-carrying conductors are associated to temperature and are unaffected by the presence of an electric potential. This is due to the typical drift velocity of electrons attributed to a current being much smaller to their thermal kinetics. In a resistor R, this can be modelled as a series voltage $v$, or a parallel current source $i$ given by

$$\overline{v^2} = 4kTR\Delta f \tag{2.8}$$

$$\overline{i^2} = 4kT \,⅟_R \Delta f \tag{2.9}$$

Where k is the Boltzmann's constant. T is the absolute temperature and $\Delta f$ is the bandwidth of the measurement in Hz. It is to be noted that thermal noise current spectral density, $\frac{\overline{i^2}}{\Delta f}$ is independent of frequency and is indeed white noise [26].

Another important noise that appear in CMOS oscillators is the flicker noise [24]. This type of noise has varied origins and is found in all active devices and also some passive devices. In bipolar transistors, for instance, it is caused by imperfections in the

crystal within the emitter base depletion layer. In the case of MOSFETs traps in the gate oxide are the cause of flicker noise[27]. These imperfections captures and release charge carriers in a random fashion and give rise to pink noise. Flicker noise is related to the flow of direct current and has a spectral density

$$\frac{\overline{i^2}}{\Delta f} = K_1 \frac{I^a}{f^b} \tag{2.10}$$

where

$\Delta f$ = a small bandwidth at frequency $f$
$I$ = direct current
$a$ = constant in range 0.5-2
$b \approx 1$

The $1/f$ frequency dependence of flicker noise is where it gets its alternative name '$1/f$ noise'. The constant of proportionality $K_1$ is an empirical parameter and typically varies by orders of magnitude in devices that have undergone same process. Consequently, flicker noise is inherently hard to predict [26].

### 2.4.2 Noise up-conversion

The noise sources presented in the previous section are wideband or of low frequency. These characteristics, in their natural state, do not directly relate to phase noises which appear close to the frequency of oscillation in an oscillator. Several literatures [20, 24, 28-31] reports how device flicker and white noise are upconverted and filtered close to the frequency of oscillation respectively. To illustrate the case of device flicker, we consider the differential ring oscillator in fig 2.22. The tail current is typically provided by a MOSFET fixedly biased into saturation and susceptible to flicker noise. Since the delay caused by each differential pairs – and hence the oscillation frequency – is dependent on the tail current, any noise in the latter modulates the 'carrier' frequency of the oscillator, $f_c$, thereby contributing to phase noise [20].

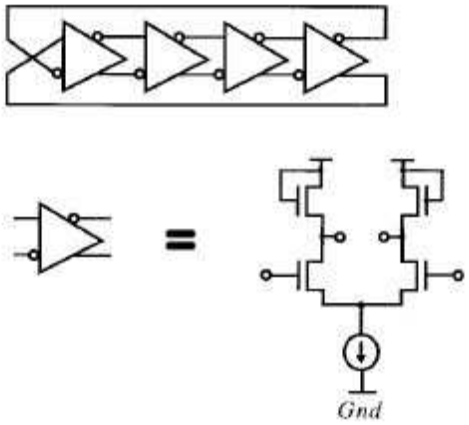



*Fig 2.22: Four stage ring oscillator.*

For the case of white noise, we consider the typical model of a harmonic oscillator affected by noise shown in fig 2.23. Keeping with Barkhausen criterion for oscillation, the loop has an infinite gain at $f_c$ and a finite gain at all other frequency. i.e. $F(j2\pi f_c)=1$ [28]. The white noise source $v_n$ within the loop sees a strong frequency selection which leads to a noise transfer function that is singular at $f_c$ and inversely declines with increasing frequency offset from $f_c$. Thus, white noise component around $f_c$ are preferentially amplified and add to the oscillation to cause phase noise.

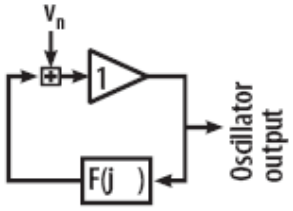



*Fig 2.23: Block diagram of feedback oscillator*

### 2.4.3 Jitter Effects on Sampling Systems

Sampling theory guaranties that the information contained in a continuous-time signal can flawlessly be recovered if the original signal has been sampled at *regular* intervals at a rate greater than twice the bandwidth of the signal. The impulse train for this sampling is actually provided by an oscillator and, as shown in preceding

sections, it is subject to timing errors. The latter affects the regularity of this impulse train and hence jitters the interval at which samples are taken.

To assess the consequence of jittered time instant, we consider a continuous-time signal $v_{in}(t)$ being sampled by a jittered clock $t = nT_s + \varepsilon_n$, where $T_s$ is the sampling period and $\varepsilon_n$ is the jitter which is assumed to follow a Gaussian distribution of $N(0, \sigma_j^2)$ [32]. The sampled signal is then $v_{out}(nT_s) = v_{in}(nT_s + \varepsilon_n)$ and the error between the input and output of the sampler at each $nT_s$ is given by

$$v_{out}(nT_s) - v_{in}(nT_s) \approx \varepsilon_n \frac{dv_{in}(t)}{dt} \tag{2.11}$$

given,

$$2\pi f_{in}\sigma_j \ll 1 \tag{2.12}$$

The error is shown to be proportional to the signal slew rate $\frac{dv_{in}(t)}{dt}$ and the jitter $\varepsilon_n$, however, the assumption in (2.12) does not hold for large $f_{in}$ in wideband systems. Kobayashi [32]showed that for any signal $v_{in}(nT_s)$ represented in its Fourier series

$$v_{in}(nT_s) = \frac{a_0}{2} + \sum_{k=1}^{\infty}\left[a_k \cos(2\pi k f_0 nT) + b_k \sin(2\pi k f_0 nT_s)\right] \tag{2.13}$$

the noise power $P_j$ due to the aperture jitter is given by

$$P_j = \sum_{k=1}^{\infty}(a_k^2 + b_k^2)\left[1 - \exp(-2\pi^2 (kf_0)^2 \sigma_j)\right] \tag{2.14}$$

This is interesting to note that

(i) $P_j$ does not depend on the sampling period $T_s$.

(ii) $P_j \le 2 \times \sum_{k=1}^{\infty}(a_k^2 + b_k^2)$ (2.15)

From the inequality (2.15) it can be deduced that for a signal with no DC component, $a_0 = 0$, $P_j \le 2 \times [\text{input signal power}]$ and hence $\text{SNR}_{\text{jitter}}$ approaches -3dB as $2\pi f_0 \sigma_j$ increases. This can be expanded to any stationary signal (i.e. a signal whose Fourier transform exists, $\int_{-\infty}^{\infty} |v_{in}(t)|\, dt < \infty$). If the Fourier transform of $v_{in}(t)$ is $V_{in}(j\omega)$ then $P_j$ is given by

$$P_j = \int_{-\infty}^{\infty} V_{in}(j\omega)^2 \left[1 - \exp(\frac{\omega^2 \sigma_j^{\,2}}{2})\right] d\omega \tag{2.16}$$

So far it has been shown that jitter causes a decrease in SNR, i.e. an increase in the noise floor. In addition to this jitter also cause accuracy and resolution lost in the form of decrease in ENOB.

The well known relationship between SNR and the resolution of the quantizer, $N$, (2.17) is often used to assess the ENOB limit due to other noise and distortion sources.

$$\text{SNR(dB)} = 6.02N + 1.76 \tag{2.17}$$

$$\text{ENOB} = \frac{\text{SNR(dB)} - 1.76}{6.02} \tag{2.18}$$

As suggested above, there is direct relationship between jitter noise and ENOB. This is graphically depicted in fig 2.24 [2].

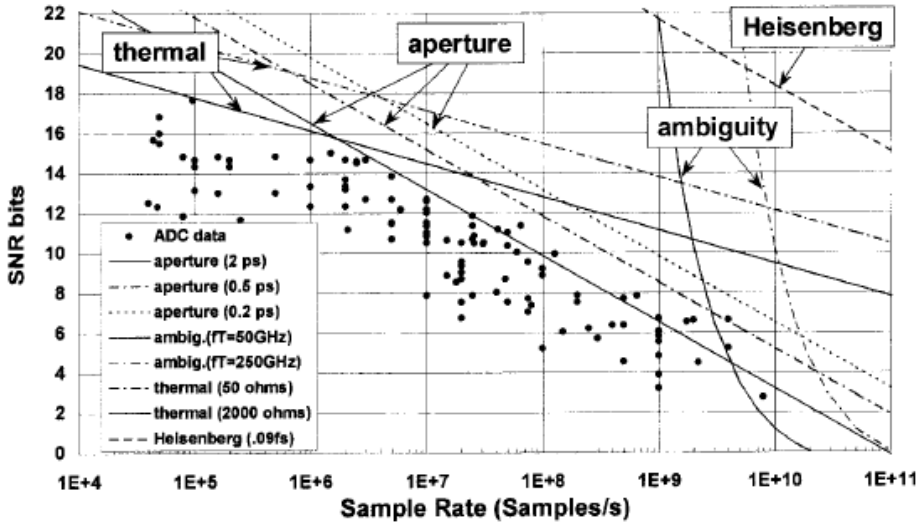



*fig 2.24: Performance limiters due to various noise and error sources.*

The input signal considered for aperture jitter in the figure above is the Nyquist frequency i.e. $f_{samp}/2$. Naturally, the Heisenberg line represents the upmost theoretical achievable resolution based of the Heisenberg uncertainty principle.

### 2.4.4 Jitter Correction

As presented so far, much effort has been done in analysing and modelling the origins and effects of jitter, in contrast, not much has been done to correct the effects of timing mismatches [25].

At tens of MHz sampling rate, bandpass filtering of the incoming clock proved to substantially reduce jitter [22, 33]. However, such studies admit high order filtering is required.

Compensation techniques have only been used to correct static jitter in time-interleaved converters. This is caused by clock path mismatches [22, 25, 34, 35]. The mismatch between each channels are estimated through an extra calibration circuit that runs in background without interrupting the conversion. The correction is then performed in digital domain through an interpolation method [25].

## 2.5 Conclusion

This chapter discussed the various figure of merits associated with ADCs and illustrated their importance. A historical background of ADCs has been reviewed pointing out the different architectures involved in data conversion. A gradual fall in resolution of ADCs with increase of sampling speed has been noted. This was attributed to jittered sampling. The sources of jitter have been discussed. It was also found that the ways to counter effect of jitter through compensation has not been investigated.

# Chapter 3

## 3. Proposed Architecture

The previous sections showed the adverse effects of jitter and the lack of proper technique to solve them. This chapter aims at providing a method of reducing the effects of jitter. The architecture consists of extracting the jitter from the clock signal by the use of phase demodulation. Since the instantaneous jitter can be derived, an approximation algorithm can be used to provide error correction to the sample. The approximation technique should provide a better estimate to the correct sample value than the jittered sample.

### 3.1 Jitter estimation

The output of an ideal oscillator can be described as

$$v_p(t) = \sin(\omega_c t) \tag{3.1}$$

In presence of coloured noise sources, such as flicker noise, the output of the oscillator is to be modified to

$$v(t) = v_p(t + \alpha(t)) = \sin(\omega_c t + \alpha(t)) \tag{3.2}$$

Where $\alpha(t)$ is the instantaneous timing noise or timing jitter. The total phase of the signal, $\Phi(t)$, and the excess phase or phase noise, $\phi_n(t)$, are given by [36-39]

$$\Phi(t) = \omega_c[t + \alpha(t)] \tag{3.3}$$

$$\phi_n(t) = \omega_c \alpha(t) \tag{3.4}$$

From the equation above it is clear that the phase noise modulates the oscillation phase. As such, phase demodulation of $v(t)$ provides a proper estimate of the instantaneous jitter present in the clock.

#### 3.3.1 Phase Demodulation

Indeed coherent detection of the phase in this scheme does not make sense. The reason behind this being that it will introduce another oscillator, which is also subjected to phase noise, as reference. Accordingly noncoherent demodulation in the form of envelope detection is to be used. From (3.2), the differential of $v(t)$ can be expressed as follows,

$$\dot{v}(t) = \omega_c(1+\dot{\alpha}(t))\cos(\omega_c(t+\alpha(t)) \tag{3.5}$$

This implies that performing envelope detection on $\dot{v}(t)$, and $\dot{\alpha}(t)$ can be obtained. $\dot{\alpha}(t)$ is the instantaneous timing error with respect to the average oscillation frequency of $v(t)$ having the last zero crossing as reference. In other words it is the instantaneous jitter of the oscillator disregarding any jitter accumulation.

## 3.2 Approximation

Given the jitter at sampling time, an approximation technique can be used to estimate the signal value at the correct sampling time. The simplest approximation technique in this case would be to use linear approximation. The performance of the latter is compared with that of third order Taylor approximation. Strictly speaking, linear approximation is same as first order Taylor approximation. Given a signal $x_i(t)$ is infinitely differentiable at some time $t = t_0 + \alpha$, the Taylor series approximation at $t = t_0$ is given by

$$\begin{aligned} x_{taylor}(t_o) = {} & x(t_o+\alpha) + \mathrm{D}(x)(t_o+\alpha)(-\alpha) + \frac{1}{2}\mathrm{D}^{(2)}(x)(t_o \\ & +\alpha)(-\alpha)^2 + \frac{1}{6}\mathrm{D}^{(3)}(x)(t_o+\alpha)(-\alpha)^3 + \mathrm{O}((-\alpha)^4) \end{aligned} \tag{3.6}$$

Where, D is the deferential operator w.r.t *t* and $\mathrm{O}(-\alpha^4)$ represents the remainder of the series.

Substituting an input signal of $x_i(t) = \sin(2\pi ft)$ into (3.6),

$$\begin{aligned} x_{taylor}(t_o) = {} & A\sin(2\pi f(t_o+\alpha)) - 2A\cos(2\pi f(t_o+\alpha))\pi f\alpha \\ & - 2A\sin(2\pi f(t_o+\alpha))\pi^2 f^2\alpha^2 + \frac{4}{3}A\cos(2\pi f(t_o \\ & +\alpha))\pi^3 f^3\alpha^3 \end{aligned} \tag{3.7}$$

Considering an input signal with frequency $f = 1.2\,\mathrm{GHz}$ and a jitter of the order of picoseconds results in a $\mathrm{O}(-\alpha^4)$ of the order of $10^{-12}$ V. This is below the typical white noise level in VLSI systems [26] and can therefore be safely neglected in equation (3.7). In this scenario, the instantaneous jitter error in the sample is given by

$$\begin{aligned} e_j(t) &= |x_i(t) - x_j(t)| \\ &= |\sin(2\pi ft) - \sin(2\pi f(t+\alpha))| \end{aligned} \tag{3.8}$$

and the error caused by the Taylor approximation can be scribed as

$$e_t(t) = \left| x_i - x_{taylor} \right| \tag{3.9}$$

$$\begin{aligned} e_t(t) = \Big| & A \sin(2\pi f t) - A \sin(2\pi f (t+\alpha)) + 2A \cos(2\pi f (t \\ & + \alpha)) \pi f \alpha + 2A \sin(2\pi f (t+\alpha)) \pi^2 f^2 \alpha^2 \\ & - \frac{4}{3} A \cos(2\pi f (t+\alpha)) \pi^3 f^3 \alpha^3 \Big| \end{aligned} \tag{3.10}$$

The worst case scenario would be that of the sampling of a sinusoid at Nyquist frequency at the zero crossings. Considering a clock frequency of 2.4GHz, the frequency of the input signal would be $f = 1.2\,\text{GHz}$. For simplicity $e_j(t)$ and $e_t(t)$ can be compared at the zero crossing at $t_o = 0\,\text{s}$. An instantaneous jitter of 1ps reduces (3.8) and (3.10) to

$$e_j(t) = 0.007539750932\text{V} \tag{3.11}$$

and

$$e_t(t) = 9.2299 \times 10^{-13}\text{V} \tag{3.12}$$

This clearly shows that the proposed approximation method reduces the jitter by several orders of magnitude at the point where the differential of the signal is a maximum. The graph figure 3.1 is a plot of (3.8) and (3.10) for the range $\alpha$=0 to $\frac{1}{4} f$ s over which $x_i(t_o + \alpha)$ is monotonic.

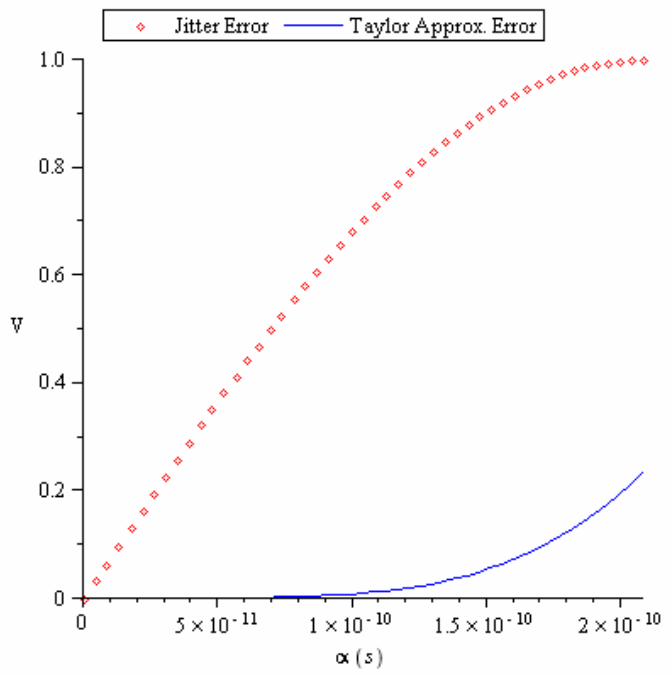



*Figure 3.1: Graph of Jitter error and Taylor approximation error versus jitter.*

Figure 3.2 shows the same graph over half cycle. Another curve is added to the graph to show the error if the Taylor series was of order 1 which is basically a linear approximation technique. Strictly speaking, the system should be beneficial if $e_t < e_j$.

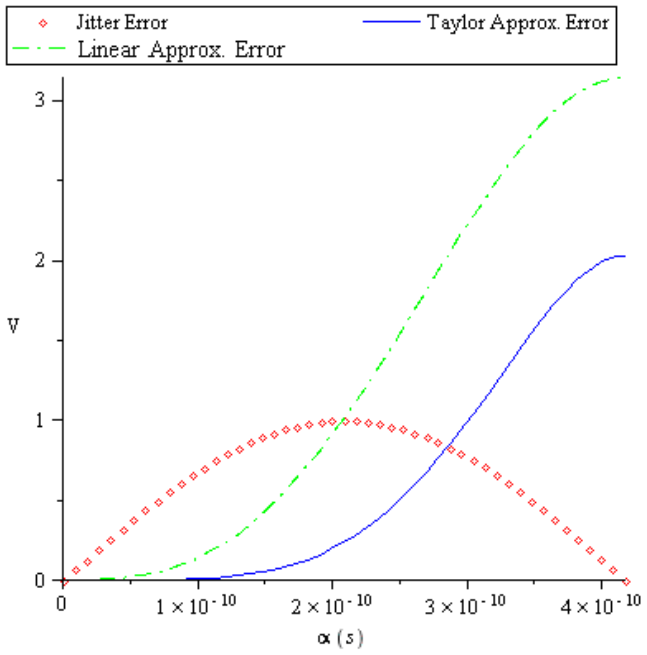
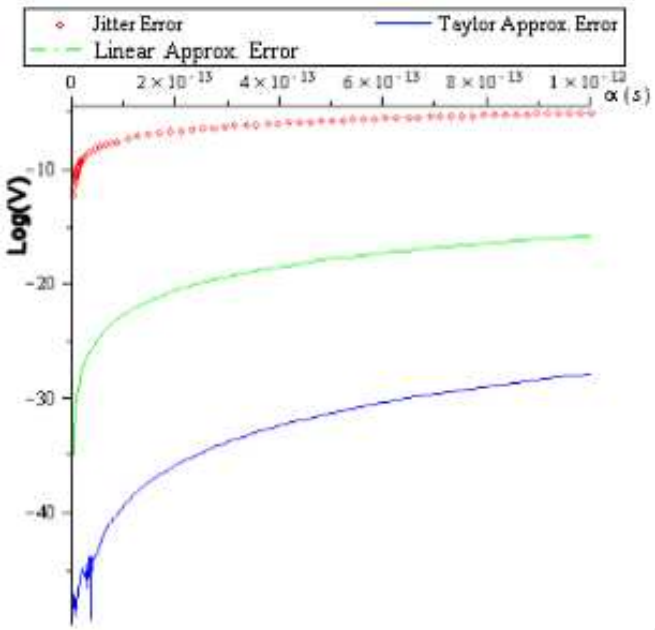



*Figure 3.2: Graph of Jitter error and Taylor approximation error and linear Approximation error versus jitter over ½ cycle.*




\

*Figure 3.3: Graph of Jitter error and Taylor approximation error and linear approximation on a logarithmic scale.*

Figure 3.3 shows that linear approximation provides a very decent amount of correction. Although superior correction could be achieved by the 3rd order Taylor approximation, its actual performance could be compromised by its complex implementation.

Altogether, the system can be summarised by the block diagram in figure 3.4. The jitter is estimated by a phase demodulator. At the same time, the input signal is differentiated and multiplied with the jitter. The result is a faithful estimate of the error in the sample value. This can be used to correct the sample.

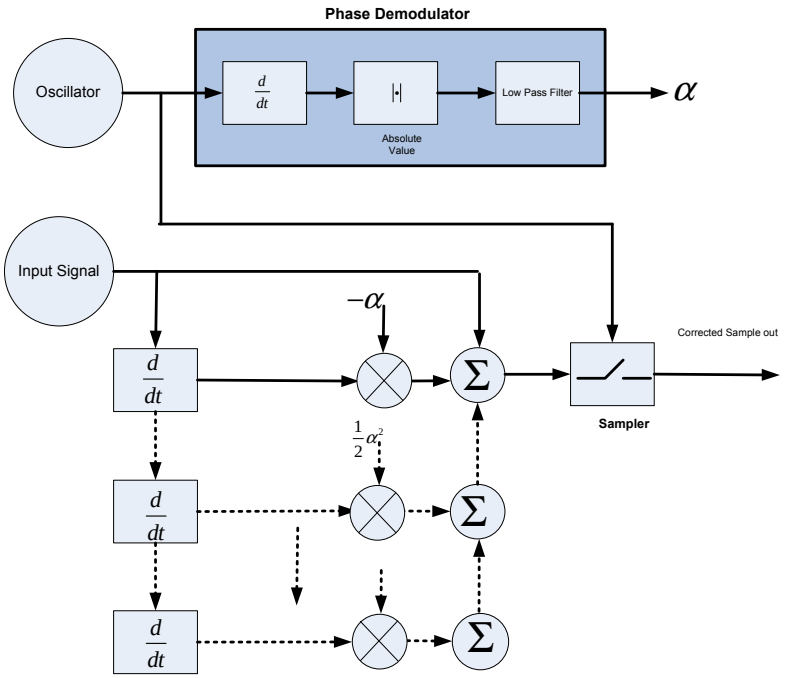



*Figure 3.4: Block diagram of the overall proposed system*

## 3.3 Simulation Results

With the aim of estimating the performance of such scheme an algorithm has been written. It is to be noted that the results shown in this section is only a theoretical estimate of the proposed scheme. It does not account for any noise or non linearity of CMOS devices.

A 1.2 GHz sinusoid is simulated on MATLAB, and sampled at a rate of 2.6 GHZ. The rms state of the art in electronics jitter is currently 0.2ps, and was the jitter standard deviation used in the simulation (rms value of a normal distribution corresponds to its standard deviation). Figure 3.5 shows the power spectral density of the original, the linearly approximated, the corrected, and the jittered signals. The first three signals differ by the order of $10^{-5}$ dB and appear superimposed on the spectrum. This clearly shows that the signal is largely unaffected by the linear approximation.

Visibly jitter has raised the noise floor. Using a measuring bandwidth of 1Hz, the SNR loss in jitter is estimated to be 12.16dB. Using Equation (2.18), this SNR degradation corresponds to a loss of 1.73 bits of resolution in the ENOB. This is

considered to be a significant improvement since the improvement in resolution was ~1.5 bits over 6-8 years prior to Walden's report [2].

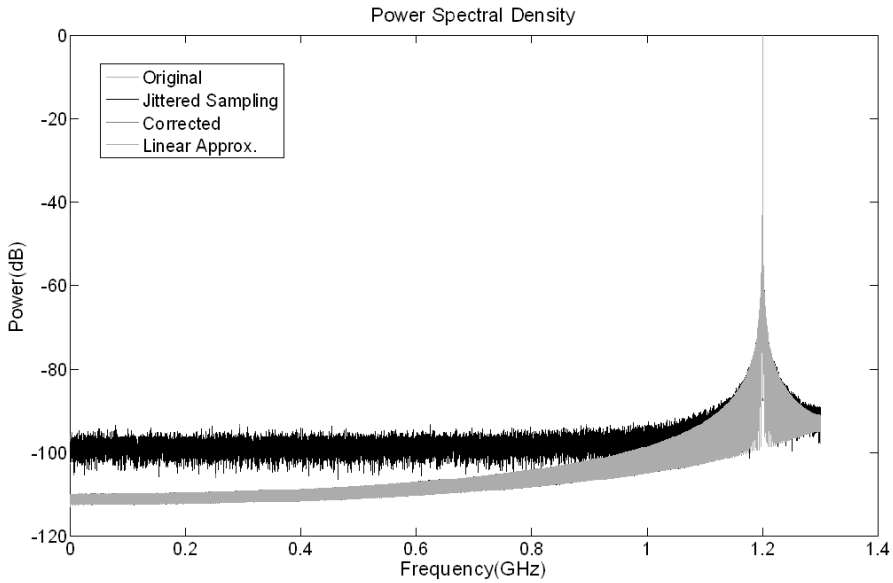



*Figure 3.5: PSD of original, jittered, corrected and linear approximated sinusoid.*

# 4. CMOS Implementation

The building blocks presented in the previous chapter provides theoretical background behind the building blocks of the architecture. This chapter aims at describing the implementation of the proposed method of correcting jitter using CMOS technology. The workings of the sub circuits involved are thoroughly enlighten. It is especially important in this research to use low noise circuits. Noisy circuits would corrupt the calculated estimate and defeat the purpose of this topology. In a bid to minimise flicker noise, a passive approach is considered as far as possible. Passive CMOS circuits are known to exhibit little or no flicker noise. Noise effects are further reduced with a differential implementation.

## 4.1 Differentiator

A differentiator can be implemented by the use of a capacitive input to an amplifier circuit, This is normally done in a fashion pictured by figure 4.1. However such implementation can produce unpredictable or erroneous outcomes at frequencies approaching the bandwidth of the amplifier. At such frequencies poles in the amplifier's transfer function causes shifts in the phase response of the differentiator. This adversely affects its performance.

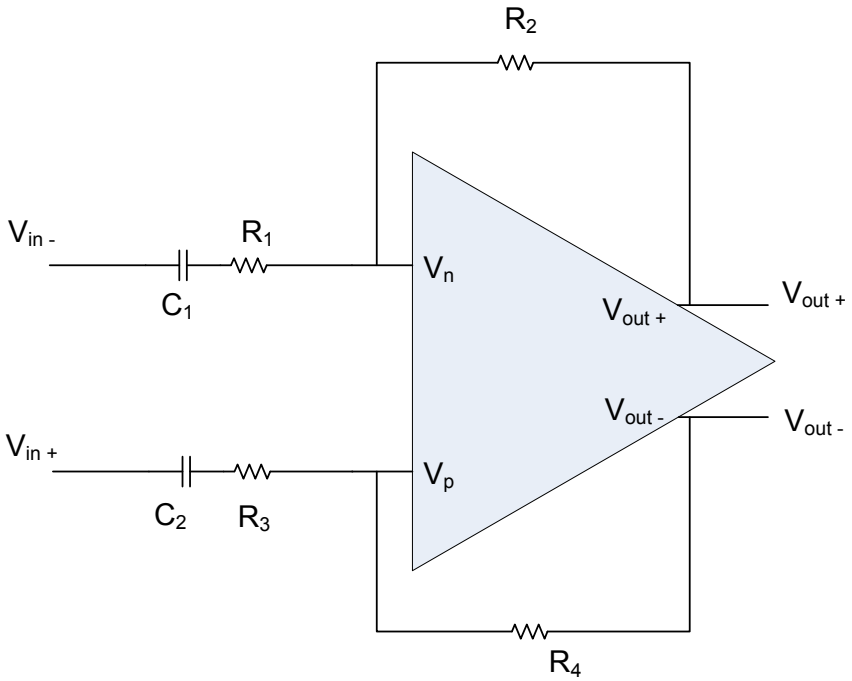



*Figure 4.1:Schematic of an active differentiator using fully differential amplifier.*

On the other hand, passive differentiator, like the on shown in figure 4.2, is designed to ensure that the phase is as close to 90° as possible throughout the spectrum of interest. This is achieved by providing a solution satisfying the under-determined set of equations in (4.1)

$$f_o = \frac{1}{2\pi RC} \qquad (4.1)$$
$$L = R^2 C$$

where $f$ is set to 2 to 3 times the max. frequency component seen at the input of the differentiator [40]. R and C has to be chosen such that L has realistic and implementable values in VLSI.

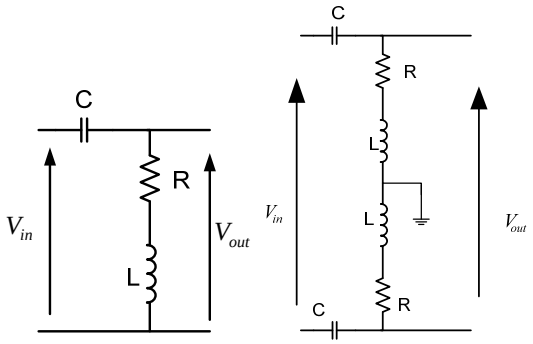



*Figure 4.2: (a)Schematic of a passive differentiator and (b) in differential implementation.*

### 4.1.1 Testing the Differentiator

The frequency and phase responses of the differentiator are shown in figure 4.3. The maximum frequency seen by the differentiator in the phase demodulator is 2.4 GHz. At that frequency the phase response is 2.2° short of the ideal 90° phase shift of differentiators. This is acceptable for this design since flicker noise has very slow dynamics compared to the oscillation.

A sinusoid of 2.4 GHz is applied to the input of the differentiator. Transient response, in figure 4.5, shows an output of 0.09V at the instants where a peak appears in the input. The attenuation caused by the differentiator is very close to -8.47dB.

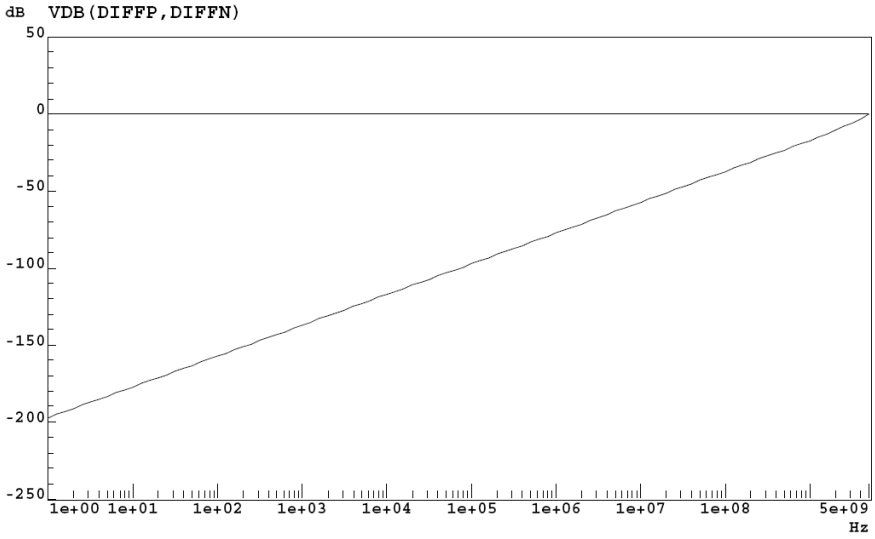



*Figure 4.3(a): Frequency response of differentiator*

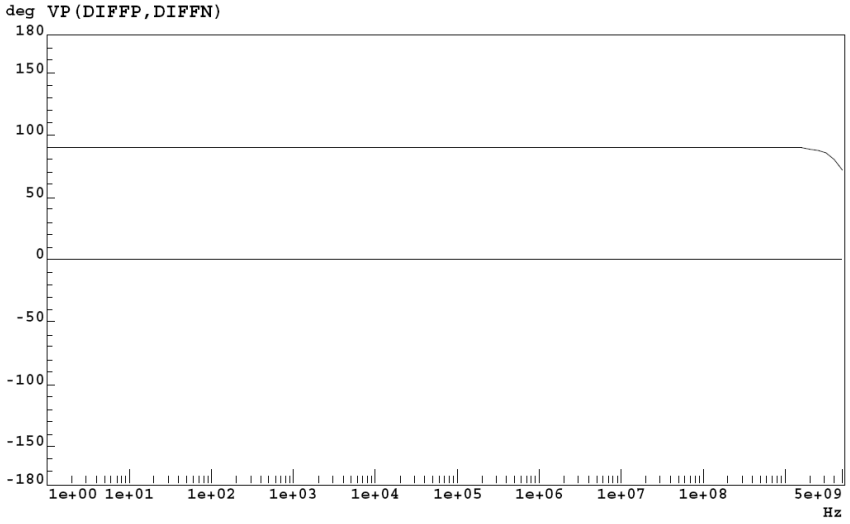



*Figure 4.3(b): Phase response of differentiator*

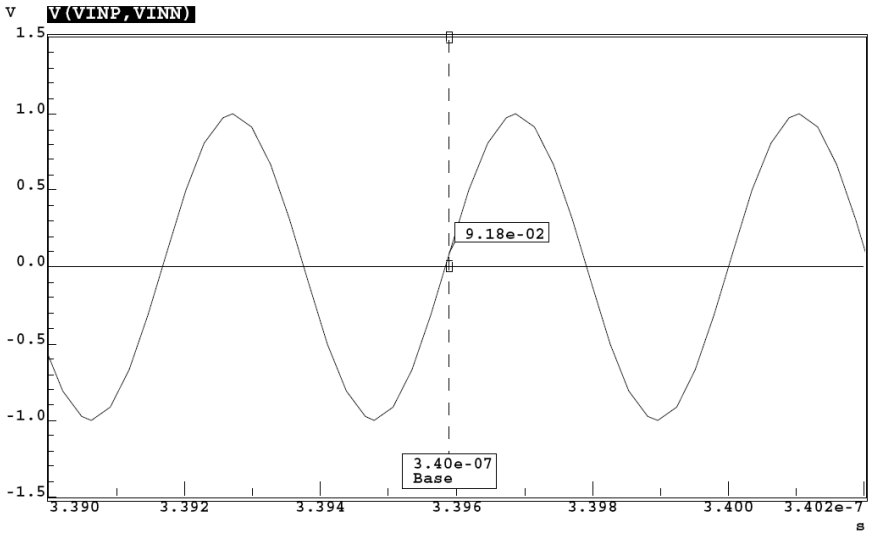



*Figure 4.4(a): Sinusoidal excitation to differentiator*

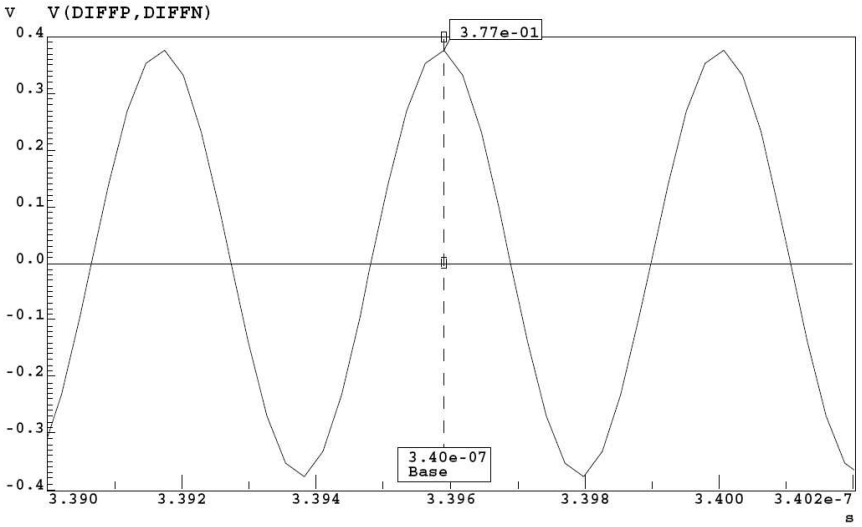



*Figure 4.4(b): Transient sinusoidal response of differentiator*

## 4.2 Mixer Design

Taylor series involves multiplication of the offset with the differential of an actual function. Here the function is the input signal to the ADC and the offset is the timing jitter. A convenient way to perform multiplication would be to use a variable gain amplifier as the one shown in figure 4.5. This design is also called a potentiometric mixer, since the input transistors act as voltage-controlled resistors. The circuit has a

very high linearity but suffers from a high noise figure [41]. This chiefly comes from thermal noise of the input transistors that get amplified.

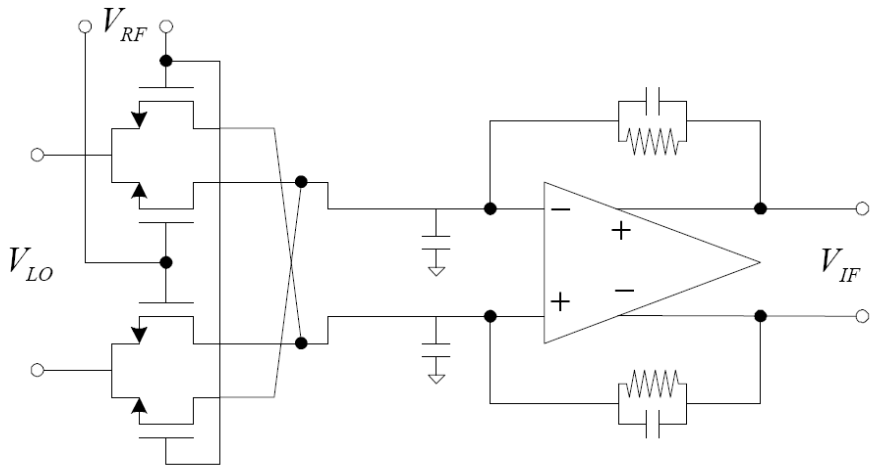



*Figure 4.5: Potentiometric mixer*

Moreover, a well known circuit that performs multiplication of signals is the Gilbert cell mixer shown in figure 4.6. The operation of this circuit is such that the input signal $V_{RF}$ modulates the transconductance of the transistors in the branches and thus provides multiplication with $V_{LO}$. Although, such an implementation is widely used in communication systems, they are admitably power hungry at high frequencies. Additionally, the transistors require DC biasing and are thus susceptible to generate flicker noise. Ways to reduce flicker in this design involve the use of large transistors higher signal power and lower frequencies, however such liberties are not always available to the designer.

Yet another design would be to keep the signal in the voltage domain and use a passive implementation instead. This involves the substitution of the tail current transistor in the active version with a capacitor feeding the cell with one of the signal. This is shown in figure 4.7. The transistors acts as time variant conductance g(t), operating in the triode region with $I_{DS} = 0$.

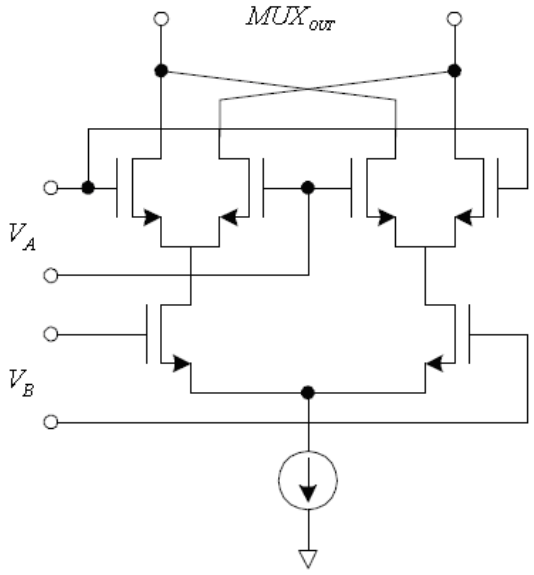



*Figure 4.6: Double Balanced Gilbert Cell Mixer*

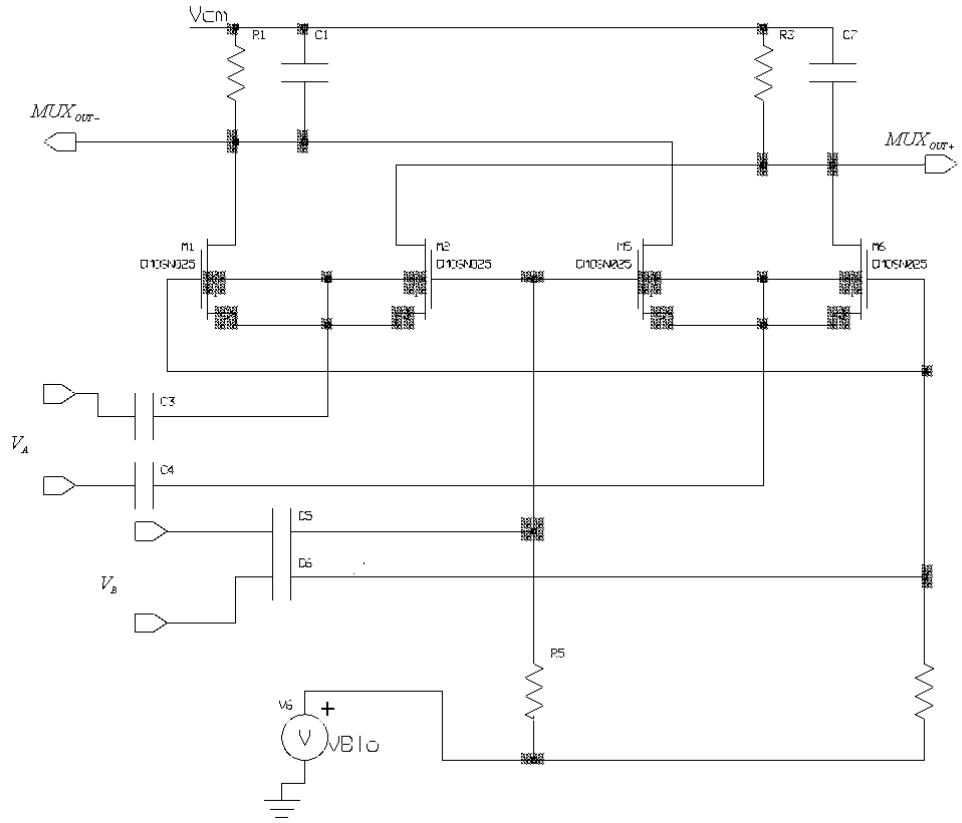



*Figure 4.7: Double Balanced Passive Mixer*

The operation of the circuit is more easily explained using its equivalent circuit in figure 4.8 with the transistors replaced by transconductances. g(t) in Siemens is given by (4.2) [42].

$$g(t) = \begin{cases} \mu_n c_{ox} \frac{w}{l} [V_B \sin(\omega_B t) - V_k] \big|_{\text{Switch on}} \\ 0 \big|_{\text{Switch off}} \end{cases} \quad (4.2)$$

where $V_k = V_{cm} + V_{th} - V_{A,DC}$

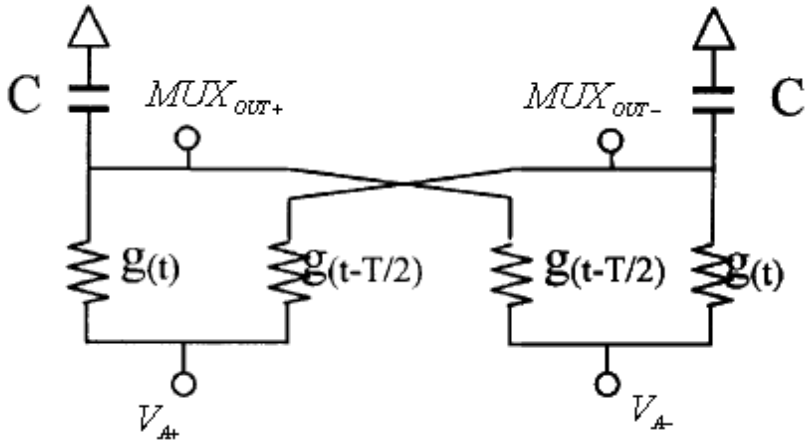



*Figure 4.8: Mixer simplified equivalent circuit*

Figure 4.9 show the transient response of the mixer when identical sinusoids of 1GHz with 1V amplitude are fed to the mixer. The results are in line with the expected $\cos^2$ output response. For this design, the signals to be multiplied are the jitter offset and the differential of the input signal. The latter can include high frequency components while the jitter is expected to have a spectrum of only a few KHz.

To further test the performance of the mixer, a 1 GHz sinusoid is multiplied by a 27 MHz sinusoid again the results, shown in figure 4.10, are in concordance with the beats expected in the output. An attenuation of 0.812 is also observed.

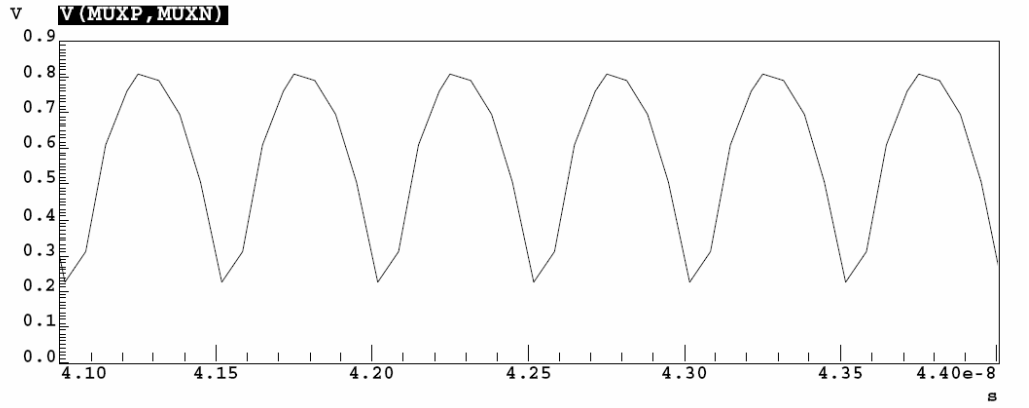



*Figure 4.9: Transient response of mixer when excited with 2 sinusoids of 1GHz*

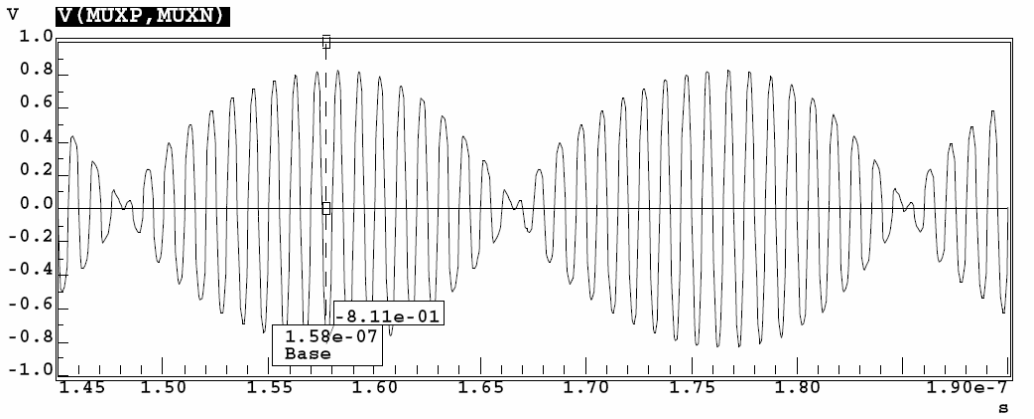



*Figure 4.10: Transient response of mixer when excited with sinusoids of 1GHz and 27MHz*

## 4.3 Oscillator Design

The oscillator used in this scheme is designed to operate at 2.4GHz. An oscillator consists of a passive part that controls the timing and an active part that provides the gain required to maintain the oscillation. The VCO is designed using the negative resistance approach as shown in figure 4.11. Clearly, $R_1$ and $R_2$ represents series resistance associated to inductors $L_1$ and $L_2$ respectively. The active part is required to provide sufficient gain to compensate for the loss provoked by $R_1$ and $R_2$. This is realized by the cross coupled NMOS that provide a negative resistance equivalent to $-(R_1 + R_2)$. Its principle of operation is worked out with the help of a test voltage $V_t$, shown in figure 4.12 [43, 44].

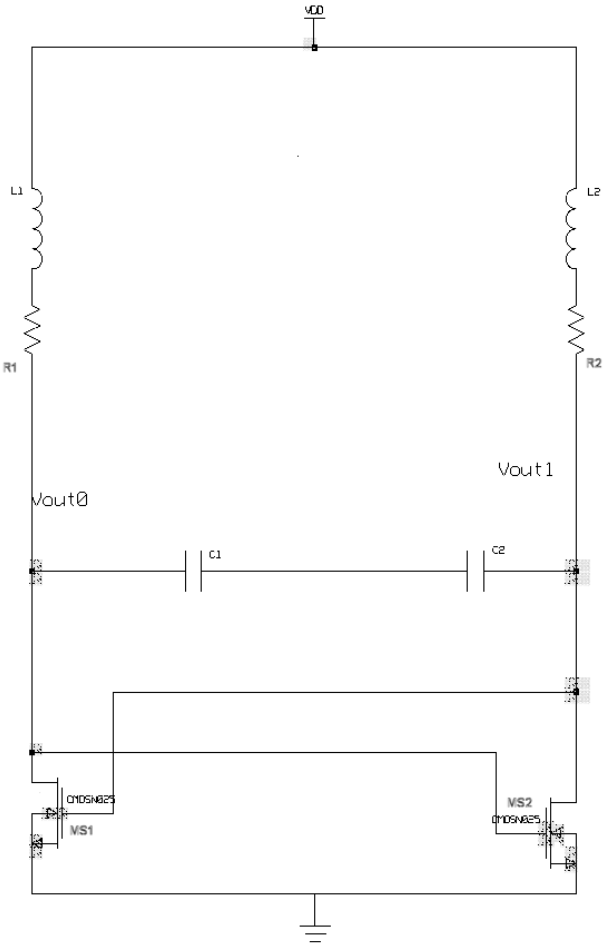
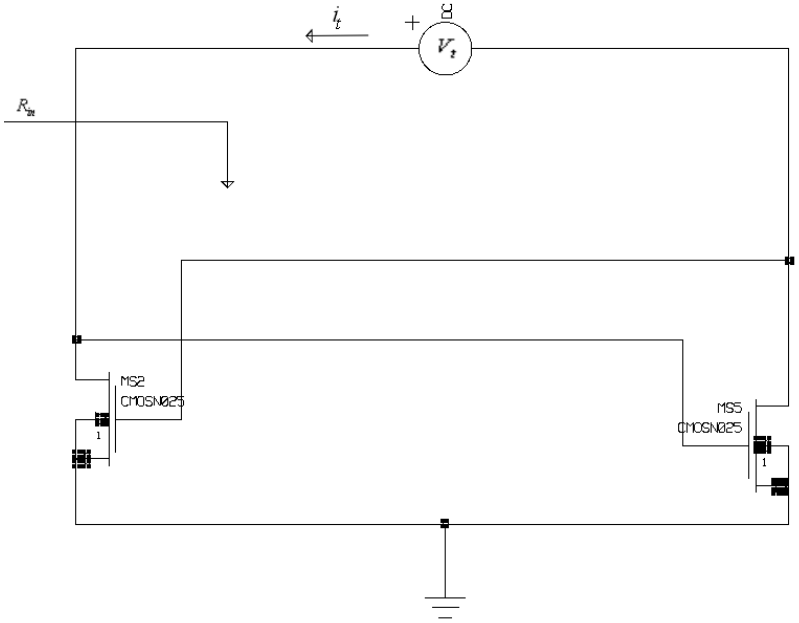
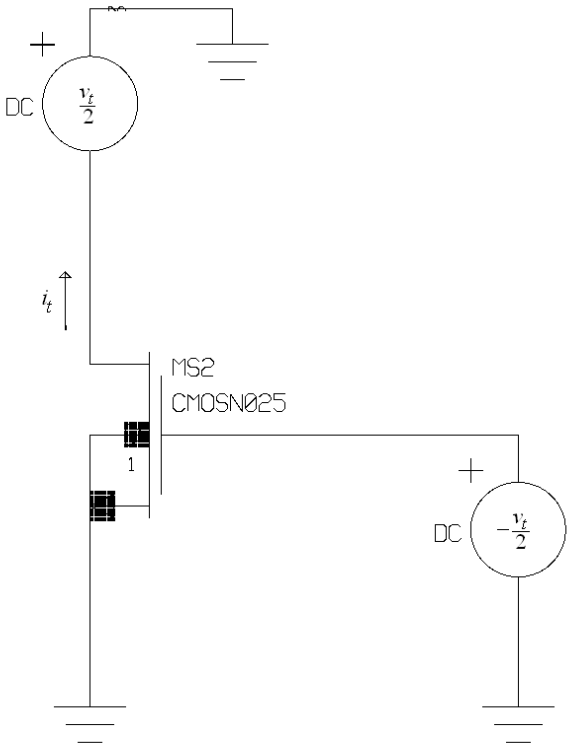



*Figure 4.11: Schematic of the VCO.*




*(a)*




*(b)*

*Figure 4.12: (a) Test circuit to determine the negative resistance. (b) Equivalent half circuit of (a).*

From a small signal stand point, the drain current $i_t$ in Figure 4.12(b) is given by

$$i_t = g_m\left(-\frac{v_t}{2}\right) \tag{4.3}$$

Accordingly, the resistance $R_{in}$ seen by the source is found to be

$$R_{in} = \frac{v_t}{i_t} = -\frac{2}{g_m} \tag{4.4}$$

where $g_m$ is the transconductance of a single NMOS.

To ensure oscillation, this negative resistance has to compensate for the loss in the equivalent parallel resistance, $R_p$, of the inductor [45]. $R_p$ is related to the series resistance, $R_s$, associated to the inductor by (4.5)

$$R_p \approx \frac{(L\omega)^2}{R_s} \tag{4.5}$$

With an inductance of 6nH and a series resistance of $1\Omega$, $R_p$ will be close to $210\Omega$.

This effectively introduces a low limit to the transconductance of the transistor. By setting $R_{in} \geq -2R_p$ in (4.4),

$$g_m \geq \frac{1}{210} \tag{4.6}$$

The frequency of oscillation, $\omega_c$, is given by

$$\begin{aligned} \omega_c &= \omega_n \sqrt{1 - g_{m0} r_s} \\ g_{m0} &= (r_s / L)C \\ \omega_n &= 1/\sqrt{LC} \end{aligned} \tag{4.7}$$

where C is the capacitance at the output nodes.
The output nodes are connected to level shifting and amplifying stage. This stage and the transistors within the oscillator add capacitance to the output nodes and have to be subtracted in the value for C in (4.7).

In order to emulate the effects of jitter in the oscillator in transient simulation, capacitors $C_1$ and $C_2$ are implemented as PMOS varactors. A sinusoid of a few KHz is fed to the varactors to jitter the zero crossing of the oscillation. The required transistor size can be calculated from the CGS0 and CGD0 spice model for the PMOS. However, the length of the transistor is made as small as possible for the given technology. This enhances the quality factor of the MOS capacitor [46]. The schematic for the oscillator is found in appendix B.

The transient analysis and spectrum of the oscillation is found in figure 4.13 and 4.14 respectively. Matlab analysis of the oscillation revealed the oscillation frequency to be 2.35 GHz. Figure 4.13 is the spectrum of the oscillation perturbed by a sinusoid of 800MHz. The effect of the perturbation is shown by a broader main lobe in the spectrum than in the previous instance. This perturbation caused an rms jitter of 34.2pS. The amount of jitter can be altered by changing the amplitude of the perturbation.

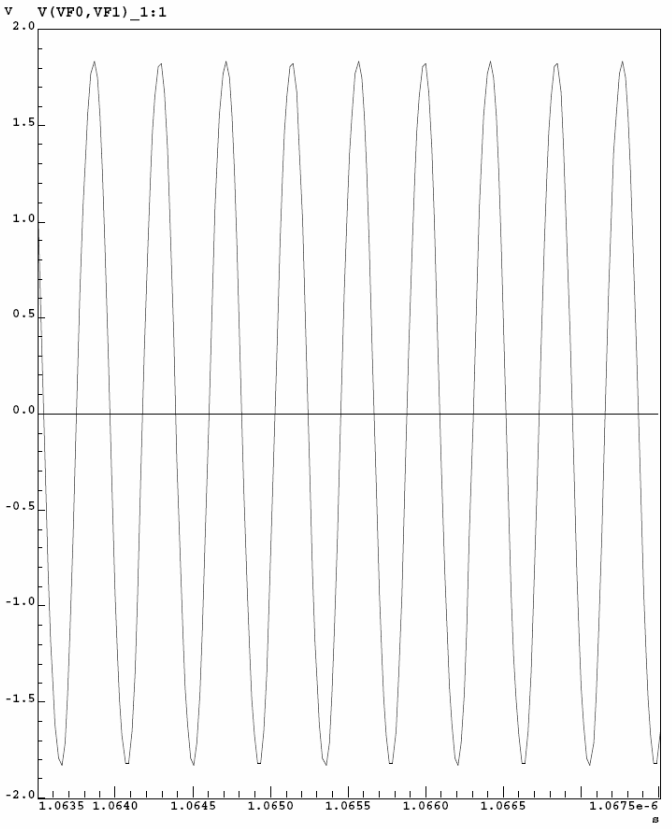



*Figure 4.13: Transient simulation of oscillator*

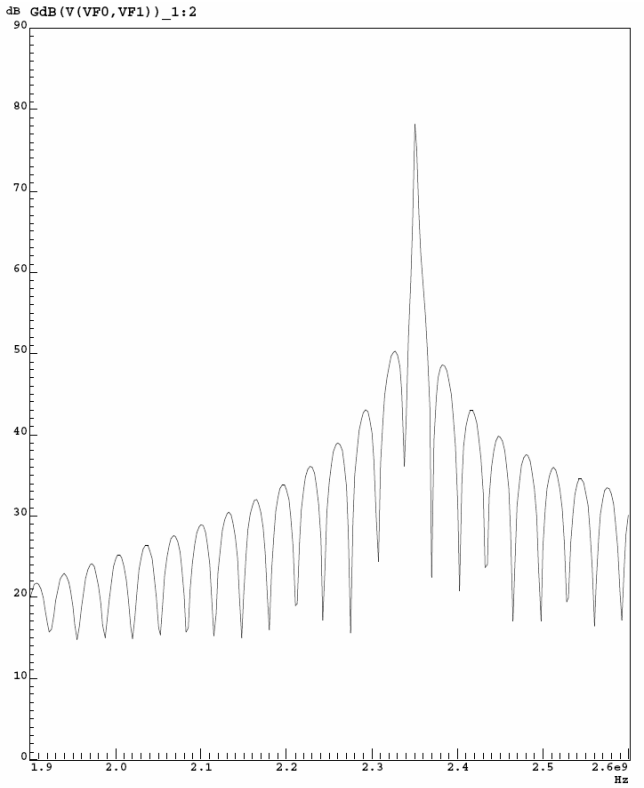



*Figure 4.14: Oscillation spectrum.*

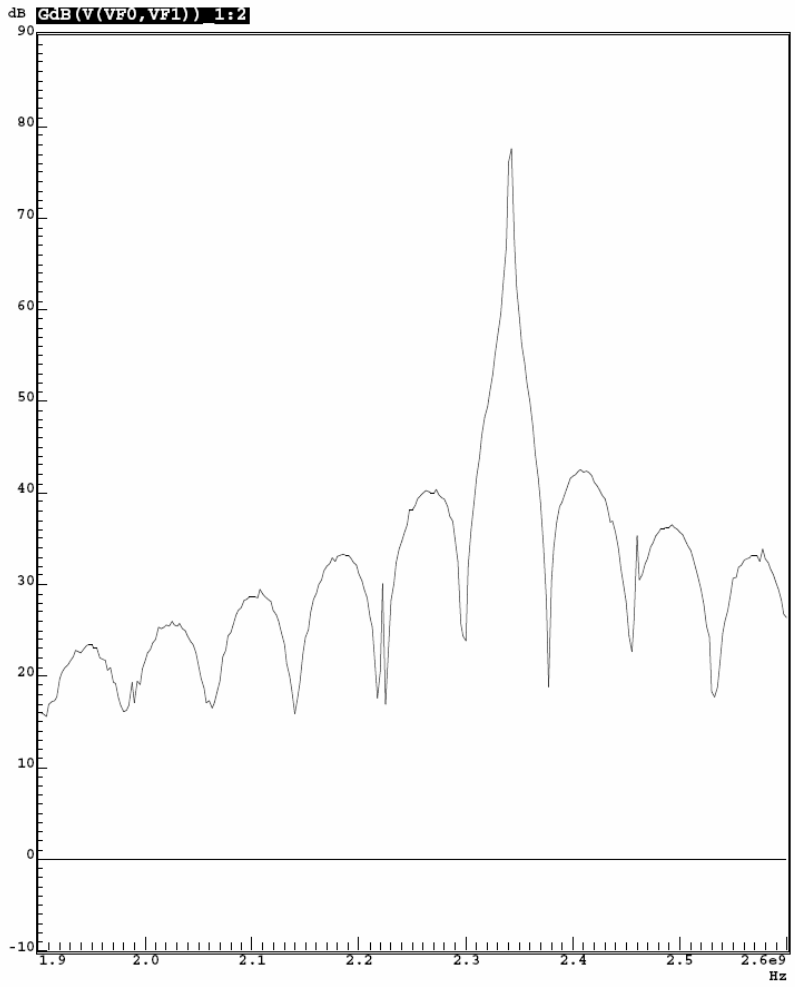



*Figure 4.15: Perturbed oscillation spectrum*

## 4.4 Phase Demodulation

Two different kinds of phase demodulation exist. They are coherent and non-coherent in nature. The former involves multiplication of the input signal with a sinusoid of known phase and same frequency as the carrier. Coherent demodulation offers superior fidelity than non-coherent type. However, in this research, the involvement of another oscillator renders this scheme inappropriate.

The scheme used involves differentiation of the oscillation followed by envelope detection. The design of the differentiator has been outlined in section 4.1. Envelope detection is implemented using a rectifier bridge and a low pass filter. The bandwidth of the low pass filter should be ideally equal to the corner frequency of the $1/f$ noise

spectrum of the oscillator. This frequency normally lies in the KHz region. This means that such a design will require large capacitors and inductors that cannot be implemented in VLSI. So, to reduce the need of external components, the implementation has to be reconsidered.

The input to the LPF has frequency components from DC to the corner frequency of the $1/f$ noise and a line spectra at twice the oscillation frequency, $2f_o$, as shown in figure 4.16. To put it simply, there is no significant energy between the $1/f$ corner frequency and $2f_o$. Hence the LPF can be designed with arbitrary bandwidth as far as it is smaller than $2f_o$.

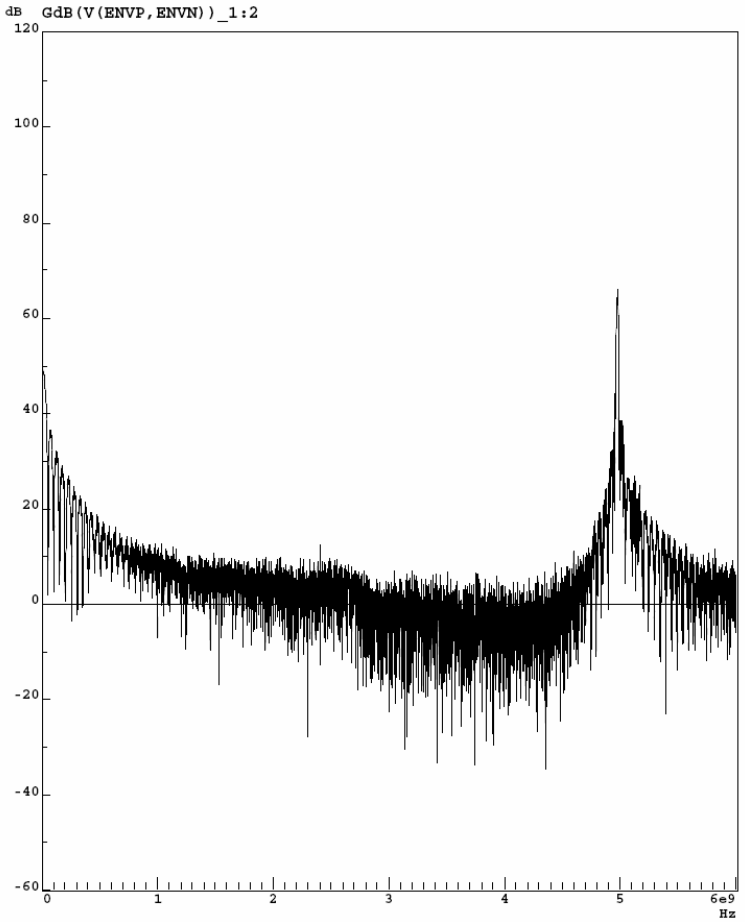



*Figure 4.16: Spectrum of the input signal to the LPF.*

## 4.5 Recommendations for a Large Bandwidth Amplifier

As stated in previous sections the mixer and the differentiator causes signal attenuation. A large bandwidth amplifier will be required in the actual implementation of this scheme to correct for the attenuation. In this simulation, the required gain has been provided by the simulator to test the topology. While the design of a low noise amplifier is out of scope of this research, a large variety of designs has been reported in literature. This section outlines guidelines for the design of a large bandwidth amplifier and implementation of fully differential amplifier circuits.

The amplifier bandwidth should be able to cater for the input oscillation frequency which is 2.4GHz in our case On chip generated noise is tackled by using a fully differential design. Thus common mode noise is reduced.

The topology recommended is a Current Mode Logic (CML), and has proved to provide superior performance at high speed (eg. [47]) compared to static CMOS inverter implementations. The latter engross complementary transistors in series. Only one transistor conducts at a time but the capacitance of both transistors are always present, in parallel, in a small signal view point. This represents a drawback in high speed circuits. Also, static CMOS inverters are sensitive to and cause significant power rail bounce. It is perhaps good to note that this is also a source of jitter [48].

On the other hand, CML buffers provide better immunity to noise. They include a current source that maintains a constant supply fairly well. The fully differential architecture makes CML buffers insensitive to supply fluctuations. However to maintain the transistors in saturation mode, thus avoiding distortions, the voltage swing of CML buffers are much less than rail to rail as CMOS inverters provides.
The amplifier consists of a number of differential stages to provide a reasonable gain while keeping the delay to a minimum. Assuming that the Drain-Body capacitance $C_{DB}$ is much less than the Gate-Source capacitance $C_{GS}$ then the optimum number of stages is given by

$$N_{opt} = \ln(X) \tag{4.8}$$

Where $X$ is the ratio of the transistor width of the last stage to the first stage [49]. In this thesis, $X = 100$ and 5 stages have been used.

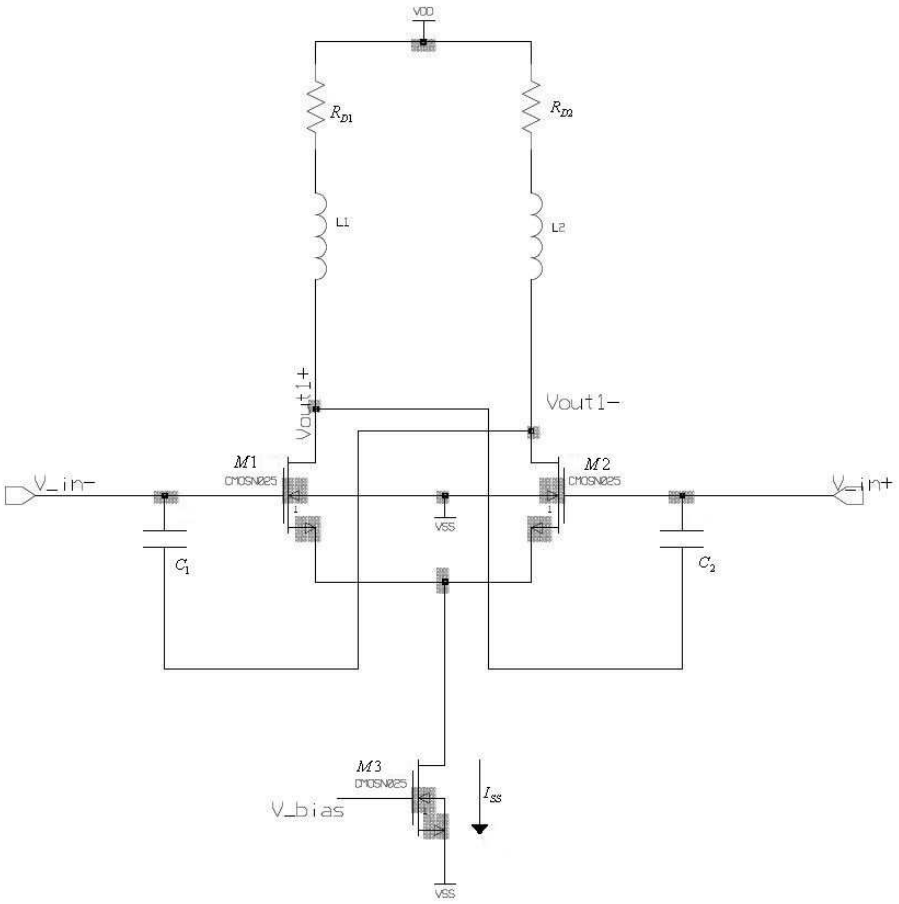



*Figure 4.17:Single stage of a CML buffer.*

Figure 4.17 shows a schematic of typical stage of a CML buffer. The 2 branches are identical and device matching is critical.

To achieve fast operation and to minimise distortion the transistors are to be kept in saturation mode. For the tail current source this implies [49]

$$V_{GS,12} + (V_{GS3} + V_{THN}) \le V_{in,CM} \le \min\left[V_{DD} - R_D \frac{I_{SS}}{2} + V_{THN}\right] \tag{4.9}$$

Where $V_{in,CM}$ is the common mode input voltage level.

Moreover, regarding the differential MOS, the output voltage is constrained to the inequality [49],

$$V_{in_{+,-},max} - V_{THN} \le V_{out_{-,+}} \le V_{DD} \tag{4.10}$$

Assuming complete current switching between the branches the output voltage is transition is from $V_{DD}$ to $V_{DD} - I_{SS}R_D$. The output swing is therefore limited to

$$V_{o,swing} = R_D I_{SS} \leq V_{THN} \tag{4.11}$$

$C_1$ and $C_2$ are neutralization capacitor. Their sole purpose is to counter the detrimental effect of the gate-drain capacitance, $C_{GD}$, that effectively couples the input and the output. In this respect, the capacitance of $C_1$ and $C_2$ should precisely match $C_{GD}$ of the NMOS [43].

To increase the bandwidth of the amplifier inductances $L_1$ and $L_2$ are added in series to the drain resistors. These actually delay the current through the resistor making more current available to charge any capacitive loads attached to the output node [43, 49]. In another view point, the inductances add a zero to the transfer function of the amplifier. The inductance is given be setting the time constants ratio $m$, given by (4.12) to 2.41 [43].

$$m = \frac{RC}{L/R} \tag{4.12}$$

Where $C$ is the lumped capacitive load that exists at the output node.

For simplicity, the input common mode voltage is set to ground. We recall that the output voltage varies from $V_{DD}$ to $V_{DD} - I_{SS}R_D$. To set the common mode of the output to ground the source follower circuit in figure 4.18 is place after the last amplifier stage.

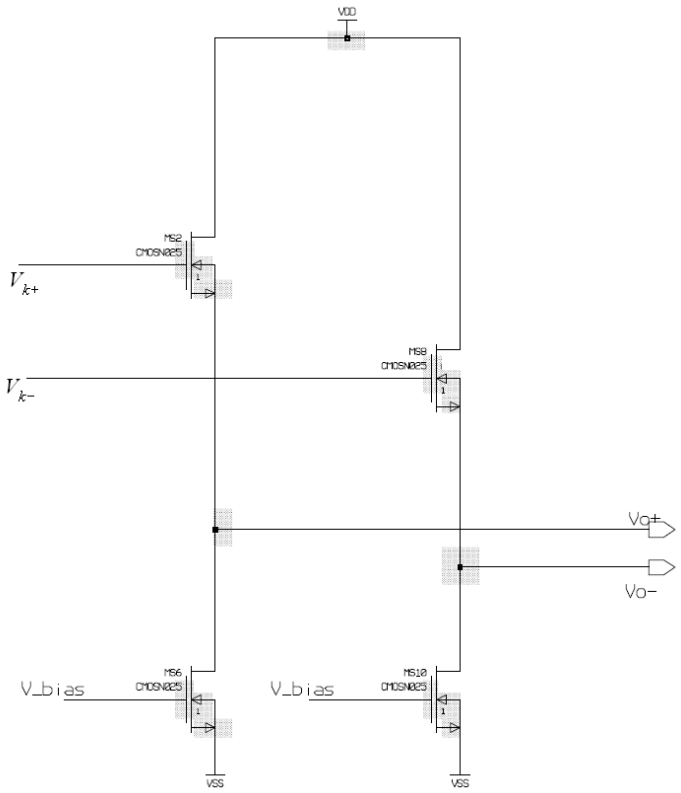



*Figure 4.18:Common Drain(source follower) stage.*

### 4.5.1 Fully-Differential Amplifier Circuits

Fully differential op-amps circuits are similar to single ended op-amps, however they are not identical [50]. Both output voltages need to be taken into consideration. Also, for proper operation and stability, both outputs require a closed loop to one of the inputs [51]. The circuit in figure 4.19 represents a fixed gain signal amplifier. The circuit is setup such that the common mode output voltage is zero. This is done by maintaining symmetry in the 2 loops (i.e. $R_1 = R_3$ and $R_2 = R_4$). In such a configuration the output is related to the input by (4.13)

$$V_{out+} - V_{out-} = \frac{(V_{in+} - V_{in-})R_2}{\frac{R_1 + R_2}{a} + R_1} \tag{4.13}$$

where $a$ is the open loop gain of the amplifier.

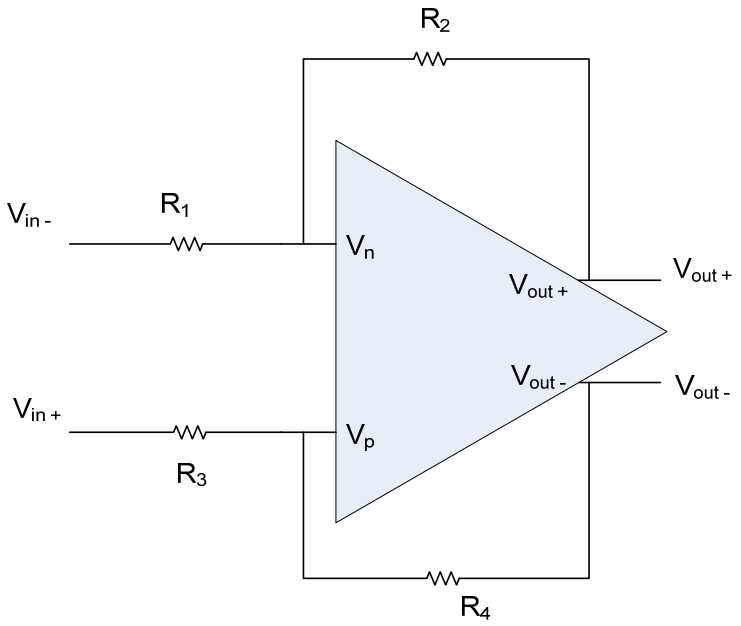



*Figure 4.19:Fixed gain closed loop fully differential amplifier.*

Specifically, when $a >> (R_1 + R_2)$ (4.13) reduces to (4.14) which is analogous to the single ended op-amp case.

$$V_{out+} - V_{out-} = \frac{(V_{in+} - V_{in-})R_2}{R_1} \tag{4.14}$$

The 6 amplifier stages presented in section 4.5 provides a gain of about 37. This is hardly enough for realistic values of $R_1$ and $R_2$. Accordingly, 3 such op-amps are cascaded to provide an overall gain of 94dB. As a result, $R_1$ and $R_2$ can be of the order of KΩ or less.

# Chapter 5

## 5. Simulation analysis and Results

Spice based simulation was performed on the completed interconnected schematic shown in figure 5.1. The MOS model is TMSC 0.25$\mu m$, made available to public domain by MOSIS. A copy of the MOS model is attached in appendix A. The Signal PLS in the schematic is an ideal sinusoidal clock of 5GHz used to sample the outputs of the envelope detector and multiplexer. This is performed using .EXTRACT commend in ELDO. A complete list of .EXTRACT command used is found in appendix B after the schematics. The simulation was run with oscillators set to 2.4972GHz and an input sinusoidal signal of 1.2GHz.

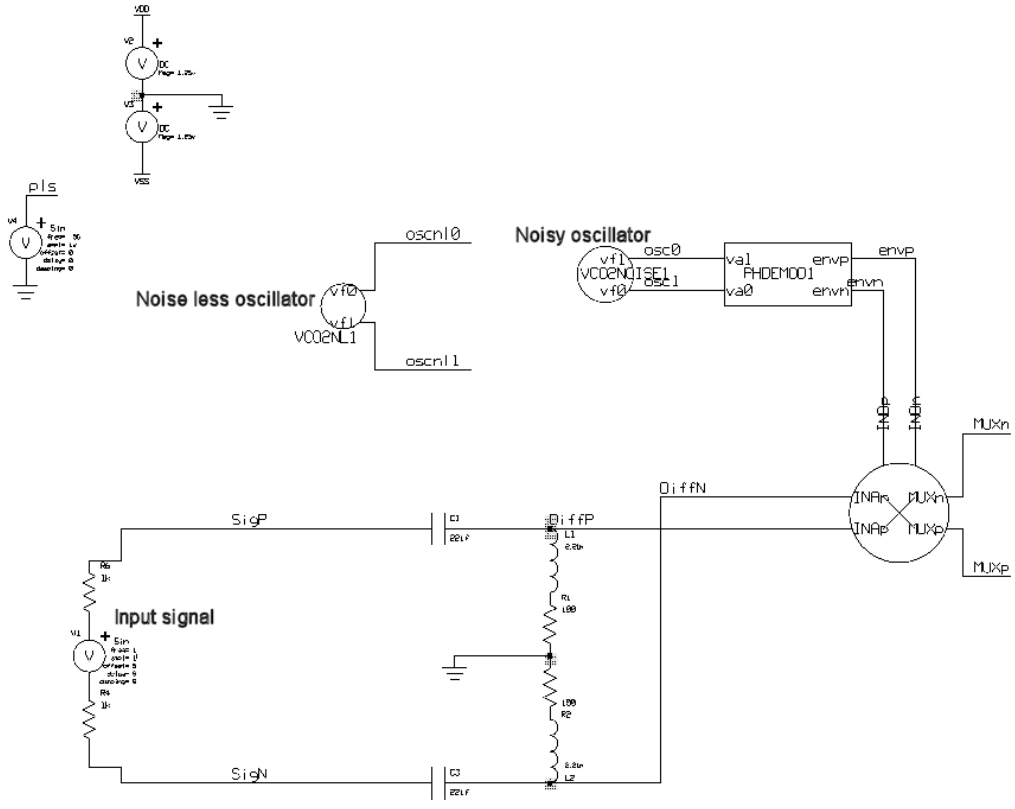



*Figure 5.1: Schematic of interconnected system*

Signal OSCNL and OSC are the noiseless and noisy oscillator signals respectively. Using both signals, the input signal SIG is sampled into a noiseless and jittered version. Next, the timing of the falling and rising zero crossings of the noisy oscillator are saved. Extracted signals are processed in MATLAB. The first few hundreds

samples are discarded since they are affected by initial transients of the circuit. This corresponds to about $0.2\,\mu S$ . Then, the jitter in the oscillation is obtained by discrete time differentiation of the zero crossing timing of the oscillator. This is easily performed using the `diff()`command in MATLAB. According to Taylor series, as established in the previous chapter the corrected signal is obtained by subtracting the mixer output with the input signal. However the mixer induces a delay in the input signal and required correction. It has been found that the delay is 13.24 nS. Because of high frequency components from the differentiator output the system is very sensitive to the delay. Severe degradation in performance is noted when the delay is altered as shown in figure 5.2.

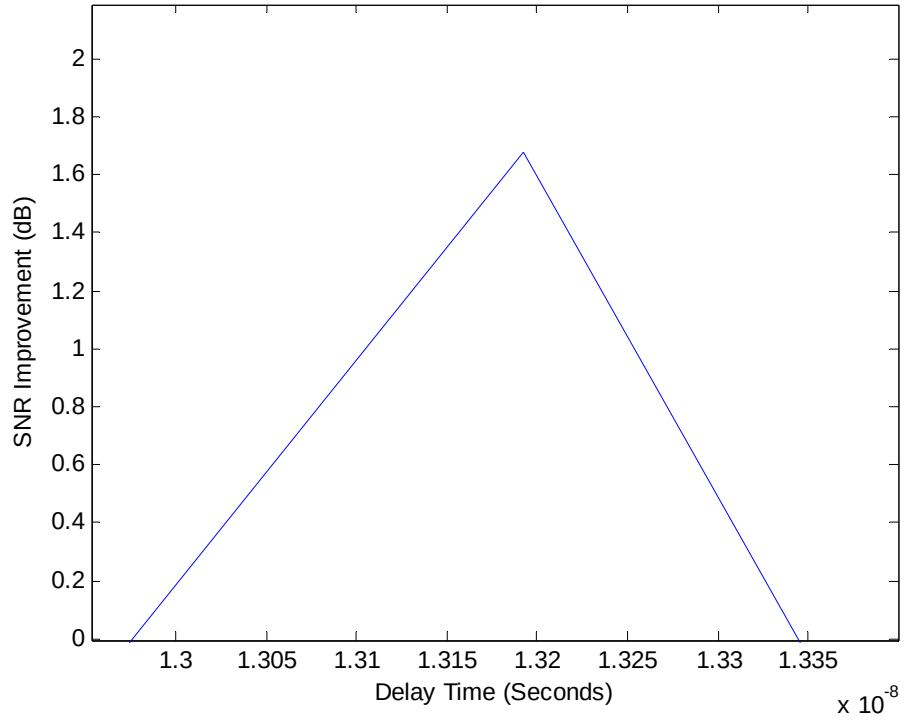



*Figure 5.2: Graph of delay correction to input signal versus improvement in SNR for a jitter of 5.1pS.*

It is a well known fact that whenever 2 signals are similar their cross-correlation and autocorrelation will also look alike. In this research the required correction signal was determine using the noisy samples subtracted the noiseless ones. This difference was cross-correlated to the output of the mixer which is in fact the correction signal under test. Notwithstanding the fact that the use of a sinusoid as noise renders the

correlation technique less meaningful, similar pattern was found in the cross correlated and autocorrelated signals. This is shown in figure 5.3. The mismatch between the 2 graph along the x-axis represent the delay involved in the correction.

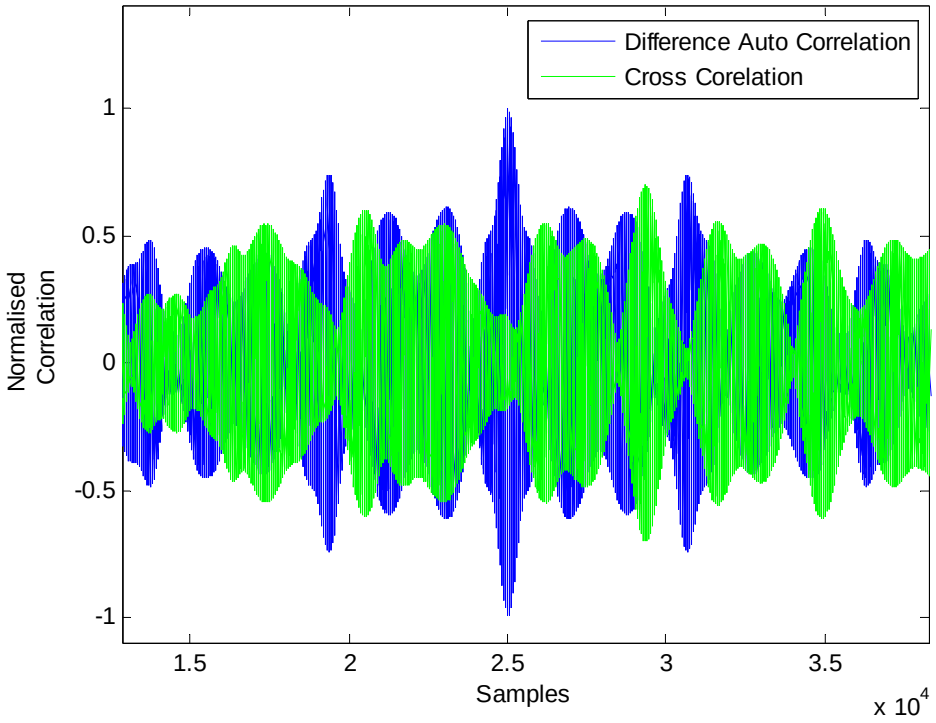



*Figure 5.3: Graph of autocorrelation of required correction signal and cross-correlation with actual correction signal from output of mixer.*

Figure 5.4 shows how the envelope detector signal modulates the output of the mixer.

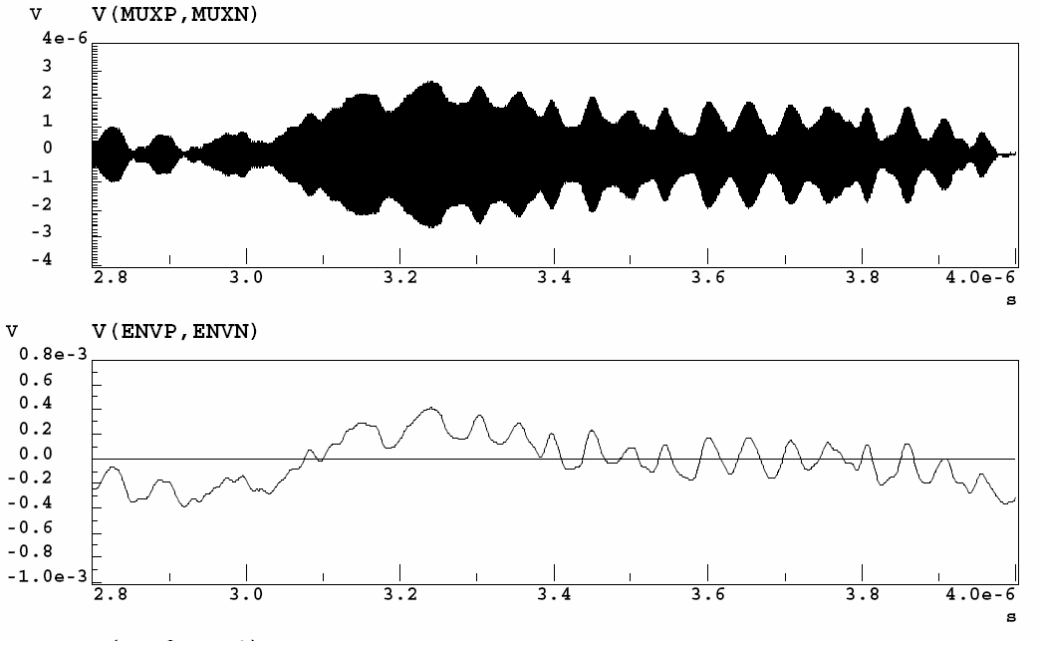



*Figure 5.4: output of mixer and envelope detector.*

## 5.1 SNR Improvement

Spectrum of noisy signals was compared to the corrected one at various jitter levels. This comparison gave an estimate of the SNR improvement resulted by the scheme. Figure 5.5 and 5.6 shows such a comparison. It is interesting to note that the SFDR remains unchanged by this scheme, but SNR is improved.

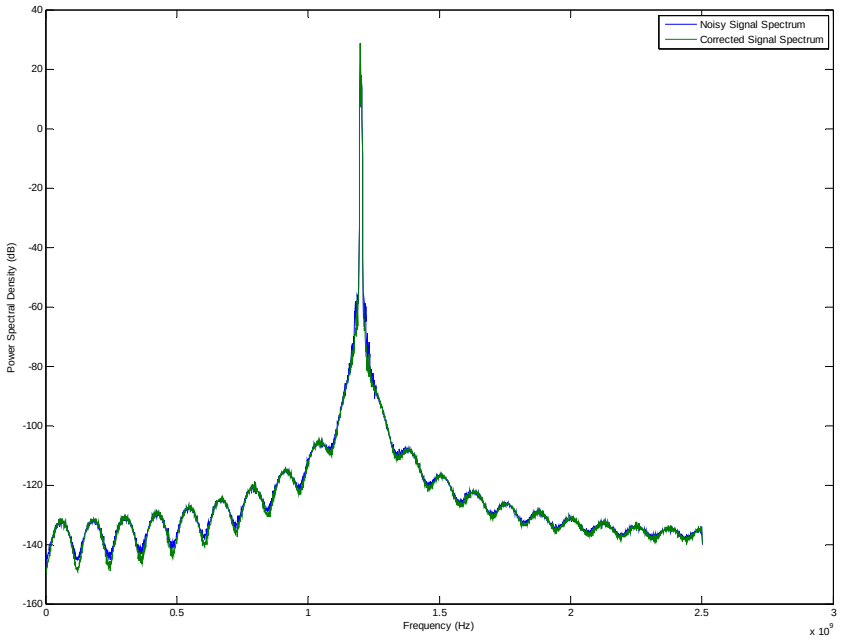



*Figure 5.5: Spectrum of corrected and noisy signals*

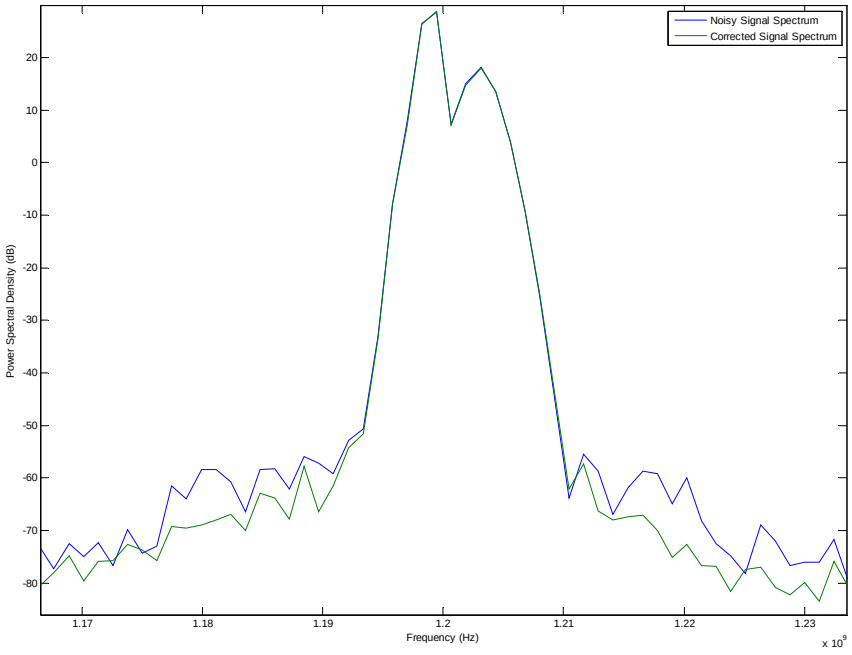



*Figure 5.6: Spectrum of corrected and noisy signals (zoom on main lobe)*

Next, two tone test was performed to check for intermodulation artefacts. Figure 5.7 shows that jitter caused first order intermodulation products. This has been reduced by about 6dB in the corrected signal.

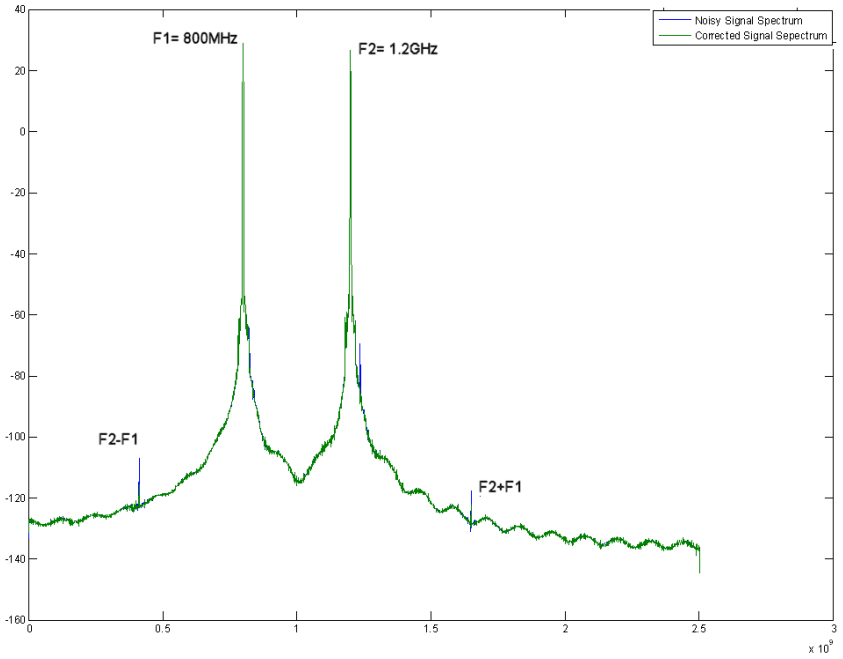



*Figure 5.7: Two tone test showing intermodulation products.*

Figure 5.8 plots the improvement in SNR observed at various jitter levels. The general decreasing trend in the benefit from the system as jitter increases is in line to theoretical predictions. Linear approximation and Taylor approximation in general, are less accurate when a wider range is involved.

According to (2.18)

$$ENOB_{increase} = \frac{SNR_{improvement}}{6.02} \tag{5.1}$$

The best improvement of 8.09dB reached was at 3.5pS rms jitter. This results in an increase in ENOB of 1.34 bits.

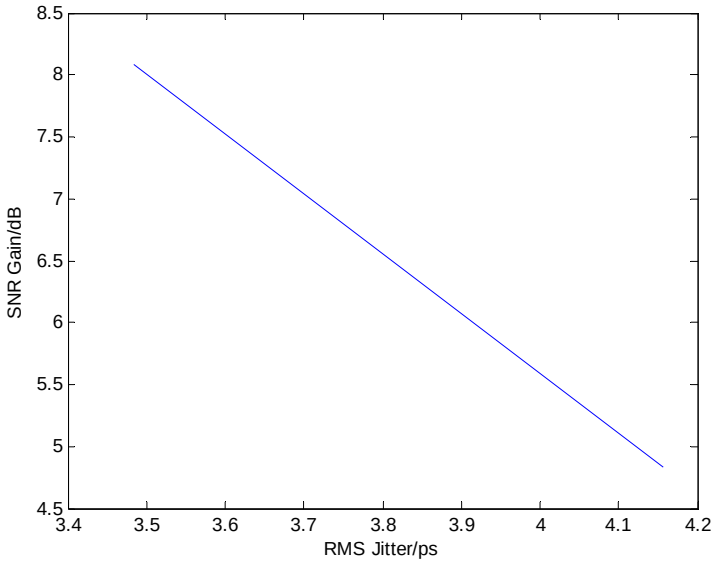



*Figure 5.8: Improvement in SNR versus Jitter.*

## 5.2 Limitations of Research

As with any simulation, several issues that sometime do not reflect real life situation are improvised. Invariably, this study faced several issues with the simulation design and analysis.

Firstly, flicker noise normally occurs at a rate of a few KHz, and consists of pink spectrum. In this research however, random noise function could not be implemented in the test environment. Rather, a 900 KHz sinusoid of different amplitude was used. While this jittered the oscillation, it did so in a periodic fashion. This however did not impede the achievement of the goal of this research.

The simulation was run on a Sun SPARC server with 4GB of RAM. The server ran out of memory if the simulation was set to run for a $T_{stop}$ of more than $5\mu S$. The sinusoid modeling the jitter had to be high enough so that several periods of the signal is present in the simulation. This explains the use a frequency as high as 900KHz.

With the oscillator setup to be jittered at 900KHz, rms jitter below 3pS could not be reached. This pushes for a real life implementation of this design.

Moreover, the .EXTRACT command in ELDO only provides figures up to 4 places of decimal. This caused a quantization phenomenon in the jitter values obtained, as shown in figure 5.8. Consequently, errors of the order of 0.1fS was introduced. This could cause minor but unpredictable effects on the results.

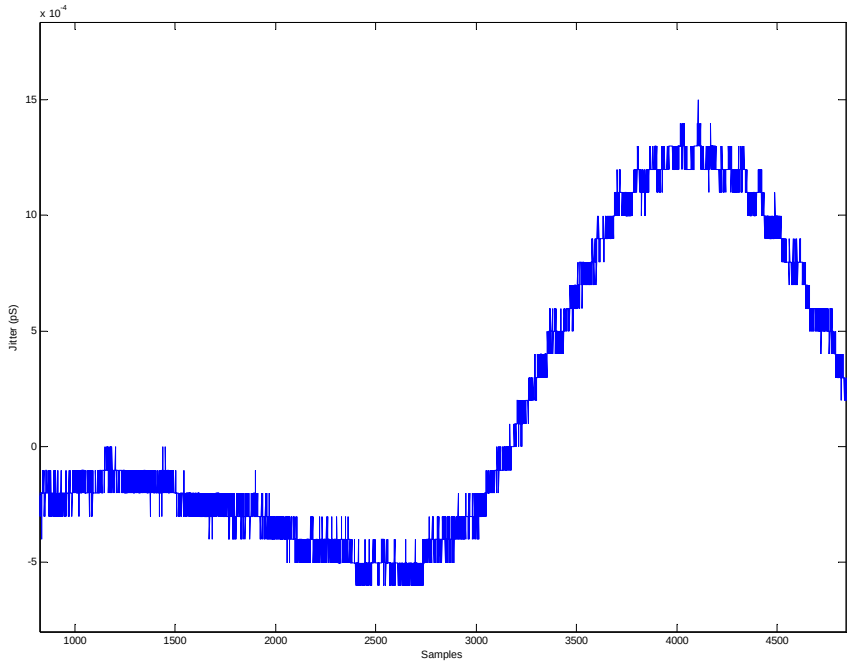



*Figure 5.8: Quantization effect on the jitter estimate due to rounding off in ELDO.*

# Chapter 6

# 6 Conclusion and Future work

This thesis presented a method of correcting jitter with passive compensating circuits. The design mainly consists of a differentiator and a mixer performing analogue multiplication and a phase demodulator. The compensation technique was based on first order Taylor series approximation. The jitter error was estimated by multiplication of the instantaneous jitter to the differential of the input signal. The instantaneous jitter was obtained by phase demodulation of the clock signal. A passive approach had been used to eliminate presence of phase noise within the compensating circuit. Simulation was performed using Mentor Graphics ELDO and TMSC $0.25\mu m$ model from MOSIS. Signals were extracted from ELDO and processed in MATLAB.

The aim of this research was to provide a compensation technique to correct the effects of jitter. Even though, the jitter simulation used was not a replica of actual jitter, this research showed that timing errors in sampling clock can be corrected by the proposed circuit. Guidelines for the design of a low noise wide bandwidth RF amplifier needed for actual implementation of such a scheme has been included. Simulation revealed an 8.09 dB improvement in SNR. Correspondingly, an enhancement in ENOB of 1.34 bits was obtained at best. No improvement in SFDR was recorded with the one tone test. On the other hand, 2-tones test showed that a non-negligible correction of 6dB was obtained in the suppression of intermodulation products.

Several issues have been outlined. However, due to limitations of the design environment, remedies could not be provided. The aim of this study has not been severely compromised by the limitations but give hindsight for future work.

## 6.1 Future work

As with any design, results and analysis of the proposed scheme tend to indicate procedures that could yield better results. These could have been considered during

the initial design, however was left out due to lack of time, budget and other relevant factors.
Several issues have been pointed out with the simulation. These were mainly simulation time, loss in accuracy and noise model used in the simulation. All of those hindrances can be remedied by an actual implementation of the proposed circuit.

Additionally on chip quantizer could be used for digitizing the signals rather than relying on an acquisition card to do this job, since sampling jitter is under investigation.

The final recommendation for future research is the design of an appropriate and accurate delay line for the purpose of realising this scheme.

# Appendix A

# A. MOS Model

*http://www.mosis.org/cgi-bin/cgiwrap/umosis/swp/params/tsmc-025/t6be_mm_non_epi_mtl-params.txt
*  RUN: T6BE (MM_NON-EPI_THK-MTL)  VENDOR: TSMC
*  TECHNOLOGY: SCN025  FEATURE SIZE: 0.25 microns
*COMMENTS: TSMC 0251P5M

*SPICE 3f5 Level 8, Star-HSPICE Level 49, UTMOST Level 8

* DATE: Jan 23/07
* LOT: T6BE  WAF: 8005
* Temperature_parameters=Default
.MODEL CMOSN025 NMOS (  LEVEL  = 53
+VERSION = 3.1  TNOM  = 27  TOX  = 5.6E-9
+XJ  = 1E-7  NCH  = 2.3549E17  VTH0  = 0.3703728
+K1  = 0.4681093  K2  = 7.541163E-4  K3  = 1E-3
+K3B  = 1.6723088  W0  = 1E-7  NLX  = 1.586853E-7
+DVT0W  = 0  DVT1W  = 0  DVT2W  = 0
+DVT0  = 0.5681239  DVT1  = 0.6650313  DVT2  = -0.5
+U0  = 284.0529492  UA  = -1.538419E-9  UB  = 2.706778E-18
+UC  = 2.748569E-11  VSAT  = 1.293771E5  A0  = 1.5758996
+AGS  = 0.2933081  B0  = -5.433191E-9  B1  = -1E-7
+KETA  = -4.899001E-3  A1  = 3.196943E-5  A2  = 0.5018403
+RDSW  = 126.2217131  PRWG  = 0.5  PRWB  = -0.2
+WR  = 1  WINT  = 0  LINT  = 1.34656E-9
+XL  = 0  XW  = -4E-8  DWG  = -1.127362E-8
+DWB  = -3.779056E-9  VOFF  = -0.0891381  NFACTOR = 1.29317
+CIT  = 0  CDSC  = 2.4E-4  CDSCD  = 0
+CDSCB  = 0  ETA0  = 6.291887E-3  ETAB  = 3.385328E-4
+DSUB  = 0.0449797  PCLM  = 1.5905872  PDIBLC1 = 1
+PDIBLC2 = 2.421388E-3  PDIBLCB = -0.0752287  DROUT  = 0.9999731
+PSCBE1  = 7.947415E10  PSCBE2  = 5.8496E-10  PVAG  = 1.01007E-7
+DELTA  = 0.01  RSH  = 3.9  MOBMOD  = 1
+PRT  = 0  UTE  = -1.5  KT1  = -0.11
+KT1L  = 0  KT2  = 0.022  UA1  = 4.31E-9
+UB1  = -7.61E-18  UC1  = -5.6E-11  AT  = 3.3E4
+WL  = 0  WLN  = 1  WW  = 0
+WWN  = 1  WWL  = 0  LL  = 0
+LLN  = 1  LW  = 0  LWN  = 1
+LWL  = 0  CAPMOD  = 2  XPART  = 0.5
+CGDO  = 4.65E-10  CGSO  = 4.65E-10  CGBO  = 5E-10
+CJ  = 1.698946E-3  PB  = 0.99  MJ  = 0.450283
+CJSW  = 3.872151E-10  PBSW  = 0.8211413  MJSW  = 0.2881135
+CJSWG  = 3.29E-10  PBSWG  = 0.8211413  MJSWG  = 0.2881135
+CF  = 0  PVTH0  = -9.283858E-3  PRDSW  = -10

```
+PK2     = 4.074676E-3    WKETA   = 7.164908E-3    LKETA   = -7.349276E-3    )
*
.MODEL CMOSP025 PMOS (                          LEVEL   = 53
+VERSION = 3.1           TNOM    = 27            TOX     = 5.6E-9
+XJ      = 1E-7          NCH     = 4.1589E17     VTH0    = -0.4935548
+K1      = 0.6143278     K2      = 6.804492E-4   K3      = 0
+K3B     = 5.8844074     W0      = 1E-6          NLX     = 6.938169E-9
+DVT0W   = 0             DVT1W   = 0             DVT2W   = 0
+DVT0    = 2.3578746     DVT1    = 0.7014778     DVT2    = -0.1881376
+U0      = 100           UA      = 9.119231E-10  UB      = 1E-21
+UC      = -1E-10        VSAT    = 1.782051E5    A0      = 0.9704347
+AGS     = 0.1073973     B0      = 2.773991E-7   B1      = 8.423987E-7
+KETA    = 0.0104811     A1      = 0.0193128     A2      = 0.3
+RDSW    = 694.5830247   PRWG    = 0.3169639     PRWB    = -0.1958978
+WR      = 1             WINT    = 0             LINT    = 2.971337E-8
+XL      = 0             XW      = -4E-8         DWG     = -2.967296E-8
+DWB     = -2.31786E-10  VOFF    = -0.1152095    NFACTOR = 1.1064678
+CIT     = 0             CDSC    = 2.4E-4        CDSCD   = 0
+CDSCB   = 0             ETA0    = 0.3676411     ETAB    = -0.0915241
+DSUB    = 1.1089801     PCLM    = 1.3226289     PDIBLC1 = 9.913816E-3
+PDIBLC2 = -1.499968E-6   PDIBLCB = -1E-3         DROUT   = 0.1276027
+PSCBE1  = 8E10          PSCBE2  = 5.772776E-10   PVAG    = 0.0135936
+DELTA   = 0.01          RSH     = 3             MOBMOD  = 1
+PRT     = 0             UTE     = -1.5          KT1     = -0.11
+KT1L    = 0             KT2     = 0.022         UA1     = 4.31E-9
+UB1     = -7.61E-18     UC1     = -5.6E-11      AT      = 3.3E4
+WL      = 0             WLN     = 1             WW      = 0
+WWN     = 1             WWL     = 0             LL      = 0
+LLN     = 1             LW      = 0             LWN     = 1
+LWL     = 0             CAPMOD  = 2             XPART   = 0.5
+CGDO    = 5.59E-10      CGSO    = 5.59E-10      CGBO    = 5E-10
+CJ      = 1.857995E-3    PB      = 0.9771691     MJ      = 0.4686434
+CJSW    = 3.426642E-10   PBSW    = 0.871788      MJSW    = 0.3314778
+CJSWG   = 2.5E-10        PBSWG   = 0.871788      MJSWG   = 0.3314778
+CF      = 0             PVTH0   = 4.137981E-3    PRDSW   = 7.2931065
+PK2     = 2.600307E-3    WKETA   = 0.0192532     LKETA   = -5.972879E-3    )
```

# Appendix B

## B. Schematics

Figure B.1 shows the oscillator used in the design. The oscillator is connected to signal conditioning circuitry.

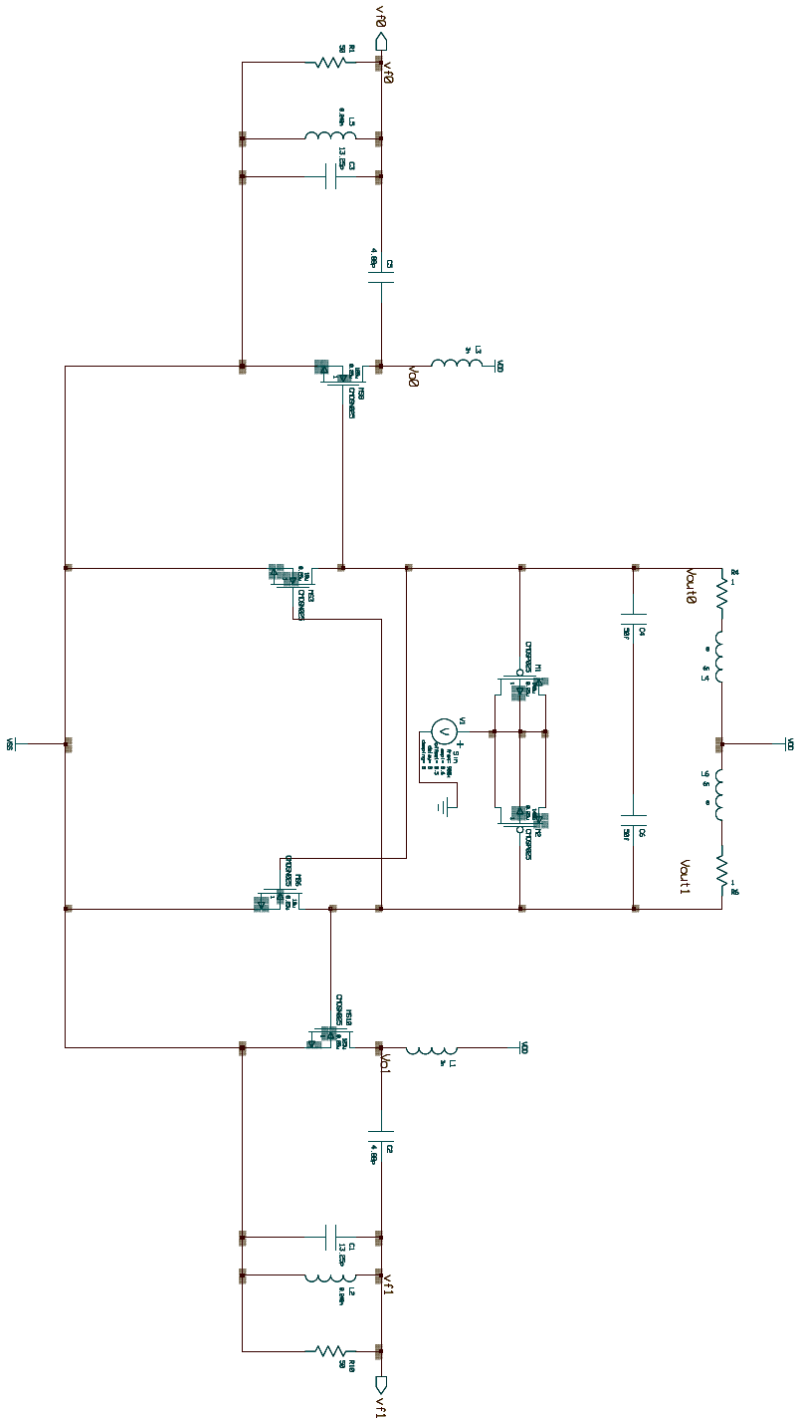



*Figure B.1: Oscillator with signal conditioning circuit.*

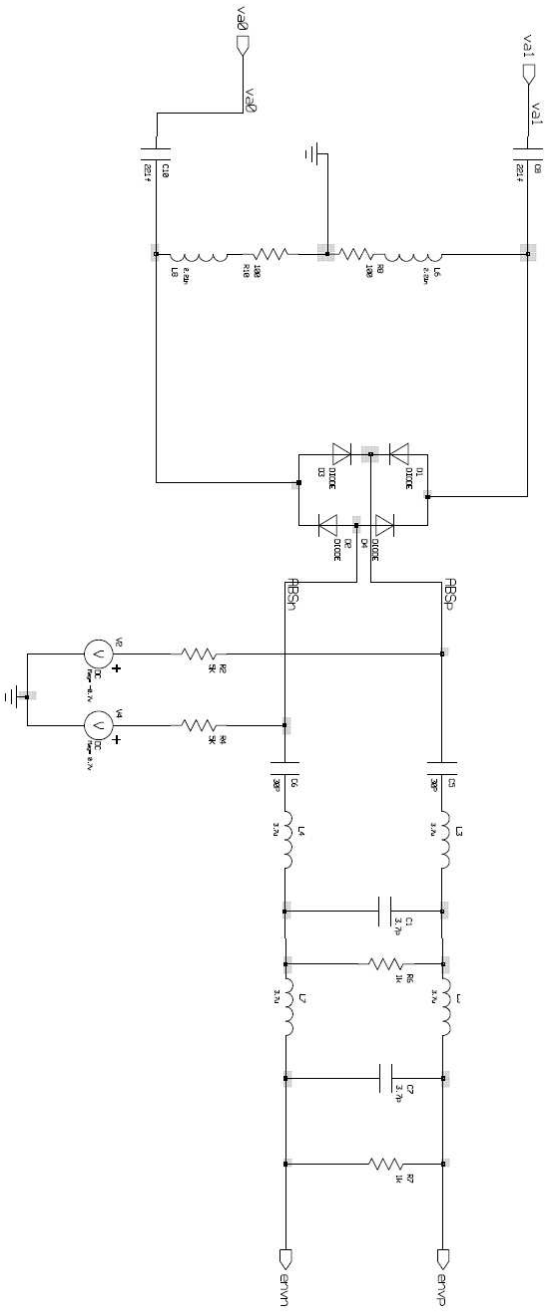



*Figure B.2: Phase demodulator.*

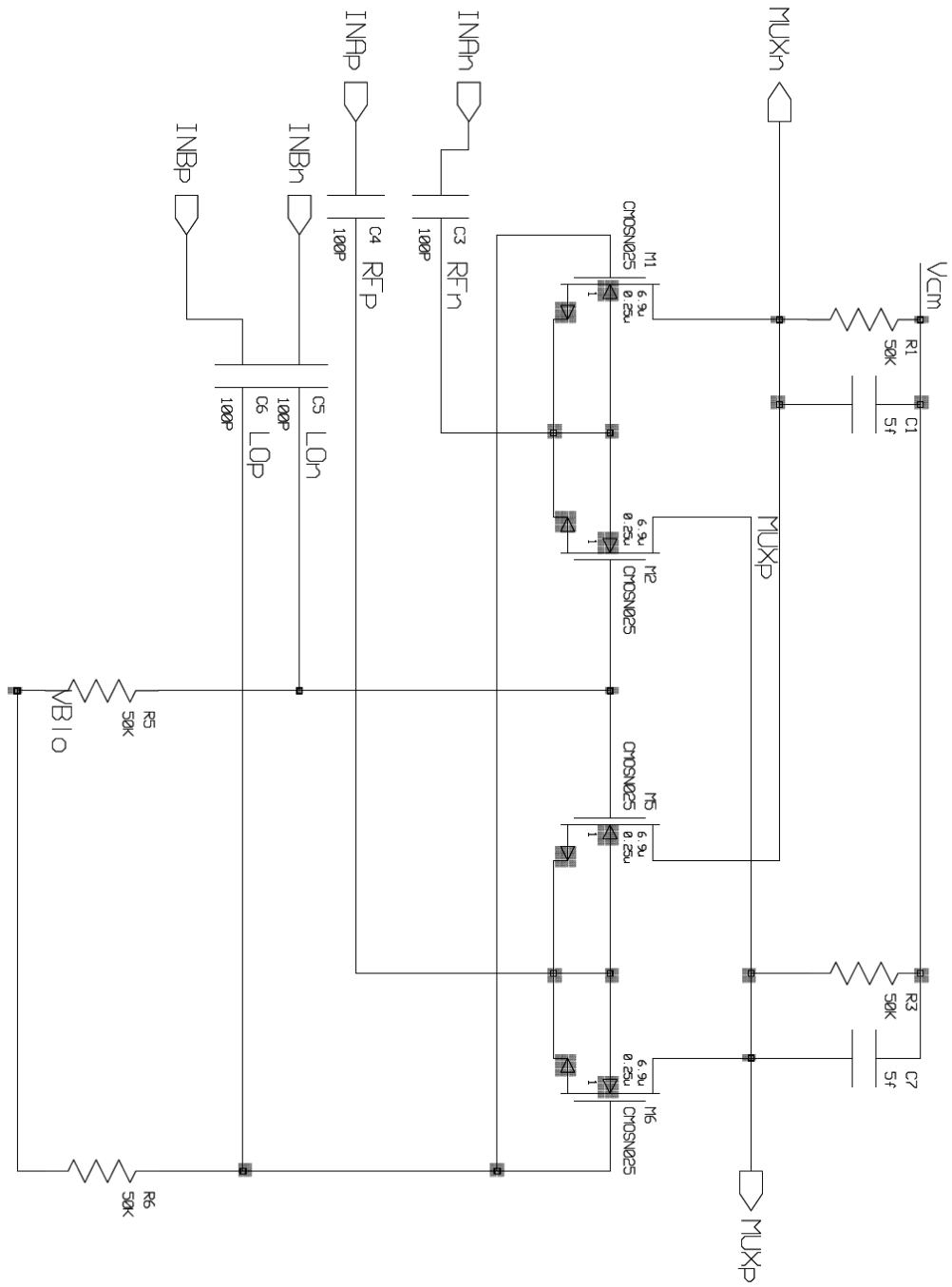



*Figure B.3: Mixer.*

The following are signal extraction commands:

```
.defwave osc=v(osc0)-v(osc1)
.defwave oscnl=v(oscnl0)-v(oscnl1)
.defwave env=v(envp)-v(envn)
.defwave sig=v(SigP)-v(SigN)
.defwave mux=v(MUXp)-v(MUXn)
*
```

```
.defmac onval(wave,clk)=yval(wave,xthres(clk,0))
*
*.extract label=osc VECT $onval(w(osc),v(pls))
.extract label=env VECT $onval(w(env),v(pls))
.extract label=oscx VECT tcross(w(osc),VTH=0)
.extract label=sig VECT $onval(w(sig),w(osc))
.extract label=signl VECT $onval(w(sig),w(oscnl))
.extract label=mux VECT $onval(w(mux),v(pls))
```

# References


[1] D. Murotake, J. Oates, and A. Fuchs, "Real-Time Implementation of a Reconfigurable IMT-2000 Base Station Channel Modem," *IEEE Communications Magazine*, vol. 38, pp. 148-152, 2000.

[2] R. H. Walden, "Analog-Digital Converter Survey and Analysis," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 539-550, 1999.

[3] W. Kester, *Analog-Digital Conversion*: Analog Devices, Inc., 2004.

[4] S. Rapuano, P. Daponte, E. Balestrieri, L. D. Vito, S. J. Tilden, S. Max, and J. Blair, "ADC Parameters and Characteristics," *IEEE instrumentation & Measurement Magazine*, pp. 44-54, 2005.

[5] "(Apr 2001) The ABCs of ADCs: Understanding How ADC Errors Affect System Performance. Maxim AN748."

[6] "(Oct 2001) Understanding Flash ADCs. Maxim AN810."

[7] C. Donovan and M. P. Flynn, "A 'Digital' 6-bit ADC in 0.25mm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 432-437, 2002.

[8] Choi and Abidi, "A 6-b 1.3-Gsample/s A/D Converter in 0.35-µm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 1847-1858, 2001.

[9] I. Mehr and L. Singer, "A 55-mW, 10-bit, 40-Msample/s Nyquist-Rate CMOS ADC," *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 318-325, 2000.

[10] "(Mar 2000) Pipeline ADCs Come of Age. Maxim AN634."

[11] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*: John Wiley & Sons, Inc., 1997.

[12] B. Le, T. W. Rondeau, J. H. Reed, and C. W. Bostian, "Analog-Digital Conversion: A review of past, present, and future," *IEEE Signal Processing Magazine*, pp. 69-71, 2005.

[13] T. Kugelstadt, "The operation of the SAR-ADC based on charge redistribution. Texas Instruments Incorporated," 2005.

[14] F. Vessal, C. Andre, and T. Salama, "An 8-Bit 2-Gsample/s Folding-Interpolating Analog-to-Digital Converter in SiGe Technology," *IEEE Journal of Solid-State Circuits*, vol. 39, 2004.

[15] H. Movahedian and M. S. Bakhtiar, "A new offset cancellation technique for folding ADC," *IEEE International Symposium on CIrcuits and Systems*, vol. 1, pp. 200-203, 2005.

[16] O. Carnu and A. Leuciuc, "Design Issues For Low Voltage, High Speed Folding And Interpolating A/D Converters," presented at The 2002 45th Midwest Symposium on Circuits and Systems, 2002.

[17] W. C. Black and D. A. Johns, "Time Interleaved Converter Arrays," *IEEE Journal of Solid-State Circuits*, vol. SC-15, pp. 1022-1029, 1980.

[18] L. Y. Nathawad, R. Urata, B. A. Wooley, and D. A. B. Miller, "A 40-GHz-Bandwidth, 4-Bit, Time-Interleaved A/D Converter Using Photoconductive Sampling," *IEEE Journal of Solid-State Circuits*, vol. 38, pp. 2021-2030, 2003.

[19] S. M. Jamal, D. Fu, N. C.-J. Chang, P. J. Hurst, and S. H. Lewis, "A 10-b 120-Msample/s Time-Interleaved Analog-to-Digital Converter With Digital Background Calibration," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 1618-1627, 2002.

[20] B. Razavi and B. A. Wooley, "Design Techniques for High-Speed, High-Resolution Comparators," *IEEE Journal of Solid-State Circuits*, vol. 27, pp. 1916-1926, 1992.

[21] B. J. McCarroll, C. G. Sodini, and H.-S. Lee, "A High-Speed CMOS Comparator for Use in an ADC," *IEEE Journal of Solid-State Circuits*, vol. 23, pp. 159-165, 1988.

[22] A. Zanchi and F. C.-Y. Tsay, "A 16-bit 65-MS/s 3.3-V Pipeline ADC Core in SiGe BiCMOS With 78-dB SNR and 180-fs Jitter," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 1225-1237, 2005.

[23] Y. Han and B. Jalali, "Continuous-Time Time-Stretched Analog-to-Digital Converter Array Implemented Using Virtual Time Gating," *IEEE Transactions on Circuits and Systems - I*, vol. 52, pp. 1502-1507, 2005.

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

[25] E. Iroaga, B. Murmann, and L. Nathawad, "A Background Correction Technique for Timing Errors in Time-Interleaved Analog-to-Digital Converters.," *IEEE International Symposium on Circuits and Systems*, vol. 6, pp. 5557-5560, 2005.

[26] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd ed: John Wiley & Sons, Inc., 1993.

[27] J. Chang, A. A. Abidi, and C. R. Viswanathan, "Flicker Noise in CMOS Transistors from Subthreshold to Strong Inversion at Various Temperatures," *IEEE Trans. on Electron Device*, vol. 41, pp. 1965-1971, 1994.

[28] A. Abidi, "How Phase Noise Appears in Oscillators," in *Workshop on Advances in Analog Circuit Design*. Italy, 1997.

[29] A. Hajimiri and T. H. Lee, "Design Issues in CMOS Differential LC Oscillators," *IEEE Journal of Solid-State Circuits*, vol. 34, pp. 717-724, 1999.

[30] A. Hajimiri, S. Limotyrakis, and T. H. Lee, "Jitter and Phase Noise in Ring Oscillators," *IEEE Journal of Solid-State Circuits*, vol. 34, pp. 790-804, 1999.

[31] T. H. Lee and A. Hajimiri, "Oscillator Phase Noise: A Tutorial," *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 326-336, 2000.

[32] H. Kobayashi, M. Morimura, K. Kobayashi, and Y. Onaya, "Aperture Jitter Effects in Wideband Sampling Systems," *Proc. IMTC*, pp. 880-885, 1999.

[33] A. Zanchi, F. C.-Y. Tsay, and I. Papantonopoulos, "Measurement and Spice Prediction of Sub-Picosecond Clock Jitter in A/D Converters," *Proc. of the 03 Int. Sym. on Circuits and Systems, 2003. ISCAS '03*, vol. 5, pp. 557-560, 2003.

[34] S. M. Jamal, D. Fu, M. P. Singh, P. J. Hurst, and S. H. Lewis, "Calibration of Sample-Time Error in a Two-Channel Time-Interleaved Analog-to-Digital Converter," *IEEE Transactions on Circuits and Systems - I*, vol. 51, pp. 130-139, 2004.

[35] H. Jin and E. K. F. Lee, "A Digital-Background Calibration Technique for Minimizing Timing-Error Effects in Time-Interleaved ADC's," *IEEE Transactions on Circuits and Systems II*, vol. 47, pp. 603-613, 2000.

[36] A. Demir, "Computing Timing Jitter From Phase Noise Spectra for Oscillators and Phase-Locked Loops With White and 1/f Noise," *IEEE transactions on Circuits and Systems*, vol. 53, pp. 1869-1884, 2006.

[37] S. Levantino, A. Zanchi, A. Bonfanti, and C. Samori, "Fast Simulation Techniques For Phase Noise Analysis Of Oscillators," *IEEE International Symposium on CIrcuits and Systems*, vol. 2, pp. 156-159, 2000.

[38] P. Vanassche, G. Gielen, and W. Sansen, "A Generalized Method for Computing Oscillator Phase Noise Spectra," *Proc. of the Int. Conference on Computer Aided Design*, pp. 247-250, 2003.

[39] V. Vasudevan, "Simulation of the effects of timing jitter in track-and-hold and sample-and-hold circuits," *Design Automation Conference, 2005. Proceedings. 42nd*, pp. 397 - 402 2005.

[40] S. Bleszynski, "Passive differentiator tops active designs," in *EDN Magazine*, 1994.

[41] C. Lackey, "Applications of quasi-floating-gate transistors for use in low-voltage closed-loop amplifier circuits." Las Cruces: New Mexico State University, 2003.

[42] S. Zhou and M.-C. F. Chang, "A CMOS Passive Mixer With Low Flicker Noise for Low-Power Direct-Conversion Receiver," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 1084-1093, 2005.

[43] T. H. Lee, *The Design of Radio Frequency Integrated Circuits*, Second Ed. ed: Cambridge University Press, 2004.

[44] N. Sneed, "A 2-GHz CMOS LC-Tuned VCO using Switched-Capacitors to Compensate for Bond Wire Inductance Variation." Berkeley: University of California, 2000.

[45] B. Razavi, "A 1.8GHz CMOS Voltage-Controlled Oscillator," *ISSCC Digest of Technical Papers*, pp. 388-389, 1997.

[46] R. Dehghani, "Design of an Optimized 2.5 GHz CMOS Differential LC Oscillator with Nonlinear Analysis for MOS Varactor," *Analog Integr. Circuits Signal Process.*, vol. 45, pp. 15-25, 2005.

[47] U. Singh and M. M. Green, "High-Frequency CML Clock Dividers in 0.13-µm CMOS Operating Up to 38 GHz," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 1658-1661, 2005.

[48] M. M. Green and U. Singh, "Design of CMOS CML Circuits for High-Speed Broadband Communications," *ISCAS*, vol. 2, pp. 204-207, 2003.

[49] P. Heydari and R. Mohavavelu, "Design of ultra high-speed CMOS CML buffers and latches," *ISCAS*, vol. 2, pp. 208-211, 2003.

[50] R. Mancini, "Developing equations for fully differential amplifiers," in *EDN*, 2003.

[51] B. Carter, "Application report SLOA064: A Differential Op-Amp Circuit Collection," Texas Instruments 2001.