

Sampling Time Error Calibration for Time-Interleaved ADCs

Nandish Mehta

Master of Science Thesis

Sampling Time Error Calibration for Time-Interleaved ADCs

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Microelectronics at Delft University
of Technology

Nandish Mehta

August 29, 2013

Faculty of Electrical Engineering, Mathematics and Computer Science · Delft University of
Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
ELECTRICAL ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of
Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled

SAMPLING TIME ERROR CALIBRATION FOR TIME-INTERLEAVED ADCs

by

NANDISH MEHTA

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE MICROELECTRONICS

Dated: August 29, 2013

Supervisor(s):

dr. Frank van der Goes

dr. Klaas Bult

prof. dr. Kofi A. A. Makinwa

Reader(s):

dr.ir.Michiel Pertijs

dr. ing. Leo de Vreede

Abstract

In this thesis the design of the timing error calibration loop for a time-interleaved ADC is described.

The realization of fully-digital radio transceiver requires a wideband capture ADC that can simultaneously capture all the commercial wireless bands present in a mobile handset. These ADCs are expected to operate at GHz sampling speed with good energy efficiency. Both of these contradicting requirements can be fulfilled by employing time-interleaved architecture. Unfortunately, time-interleaved ADCs suffer from interleaving issues like mismatch in sampling time error. These issues can be addressed by designing a dedicated calibration loop.

In this thesis an attempt is made to design a calibration loop that detects and corrects the sampling time errors with high precision. The timing error detection technique relies on introducing two additional reference ADCs. The correction of timing errors is done using the least mean square iterative algorithm (LMS). Convergence and stability of such calibration loops are extremely critical. Hence, they are exhaustively investigated in this thesis. Factors that hamper the loop convergence were identified and relevant solutions are applied to overcome them. Furthermore, it was found that the loading effect of the reference ADCs greatly affects the accuracy of the timing error detection. A simple solution using delay lines is shown to remove this effect. Finally, techniques like inserting dummy sampling circuits, scaling sampling capacitance, and matching the clock-paths, are employed to achieve timing error correction accuracy in the order of 5fs level.

Some of these techniques are implemented at the architecture level, whereas some are implemented at circuit level. The effectiveness of the architecture level techniques is verified through MATLAB modeling while the circuit level techniques are verified through circuit simulations. The sub-blocks for the calibration loop are designed in industrial 28nm CMOS process and relevant simulation results are presented. Circuits like 11-bit $10\mu\text{W}$ DAC with 0.6LSB DNL, a track-and-hold with HD_3 of 72dB at 1GHz input frequency, and clock-path with a mean delay of 11ps, are designed for the timing error calibration loop.

Keywords: LMS calibration loop, Timing error detection, Time-interleaving, Observer Effect, Reference lanes, Wideband capture ADC, Digital-to-Analog converter, Track-and-hold, Low-power.

Table of Contents

1	Introduction	1
1-1	Motivation	1
1-2	Basics of Time-Interleaved ADC	4
1-3	Application: Wideband Capture ADC	5
1-4	Target Timing Error Correction Accuracy	7
1-5	Research Goal and Contributions	8
1-6	Thesis Organization	9
2	Sampling Time Errors in Time-Interleaved A/D Converters	11
2-1	Basics of Time-Interleaving	11
2-2	Types of Interleaving Issues	12
2-2-1	Offset Mismatch	12
2-2-2	Gain Mismatch	13
2-2-3	Timing Mismatch	16
2-2-3-1	Impact of Timing Error	17
2-2-3-2	Sources of Timing Mismatch	18
2-2-4	Bandwidth Mismatch	20
2-3	Timing Error Detection and Calibration	20
2-3-1	Use of common sample-and-hold.	21
2-3-2	Foreground vs background calibration.	21
2-3-3	Digital detection and digital correction of timing errors.	22
2-3-4	Digital detection and analog correction of timing errors.	23
2-4	Summary	24

3	Timing Error Calibration Loop	25
3-1	Principle of Operation	25
3-2	Description of Calibration Loop	28
3-3	Convergence of the Calibration Loop	29
3-3-1	Input Signal Statistics	29
3-3-2	Gain mismatch and offset of the two reference lanes	30
3-3-3	Limit on Speed of Convergence	32
3-4	Stability of Calibration Loop	33
3-5	Impact of Finite Quantization of the two Reference Lanes	34
3-6	Summary	36
4	Sampling-Time Error Due to Observer Effect of Reference lanes	37
4-1	Observer Effect: Simple RC Model Analysis	38
4-2	Observer Effect: Sampling Instance Interactions	40
4-3	Isolating the Sampling Interactions using a Wire Delay	42
4-4	Mismatch between Dummy lane and REF lane	44
4-5	Summary	50
5	Circuit Implementation	51
5-1	System-Level Design	51
5-2	Clock Path	53
5-2-1	Clock-Phase generator	53
5-2-2	Sampling Edge Tuning Circuit	53
5-3	Track-and-hold Design	56
5-4	HOLD Buffer Design	59
5-5	Digital-to-Analog Converter (DAC)	61
5-5-1	Estimating Dynamic Range	64
5-5-2	Circuit Implementation	64
5-5-3	Simulation of DNL	67
5-5-4	Design Summary	67
5-6	Summary	70
6	Conclusion	71
6-1	Problem Definition: A Recap	71
6-2	Thesis contribution	71
6-3	Future Work	73
A	Target Specifications	75
B	MATLAB Code to Simulate Basic Interleaving Issues	77
C	Simulation Setup for Timing Errors	81
C-1	Coherent FFT	81
C-2	Setting Up Variables	81

List of Figures

1-1	Mitola's wideband radio [1]	2
1-2	FoM versus ENOB and sampling frequency for ADCs published at ISSCC and VLSI from 1997-2013 [2]	3
1-3	FoM versus sampling frequency for single lane and M-lane interleaved ADC	4
1-4	Block diagram of basic 4-ADC interleaved with timing error.	5
1-5	Typical Rx link for zero-IF receiver.	6
1-6	Wideband capture receiver architecture for future mobile applications.	6
1-7	Spurious free dynamic range requirement for wideband capture ADC for GSM signals.	8
2-1	Block Diagram of an ideal 2x interleaved ADC with timing waveforms.	12
2-2	Input and output signal spectrum of ideal 2-interleaved ADCs	13
2-3	Effect of input offset and its mismatch on the output spectrum of 4-interleaved ADCs for sinusoidal input tone of $f_{IN} = 498MHz$ sampled at $f_S = 5GHz$	14
2-4	Spurs due to gain error in the output of a 4x interleaved ADCs	15
2-5	Error in sampling input signal due to timing error	17
2-6	Spectra of input signal and timing error component.	18
2-7	Bound on timing error for varying SNR and input frequencies	19
2-8	Typical track-and-hold circuit for 2x interleaved ADC	20
2-9	Impact of bandwidth mismatch on SNR for a 2x-interleaved ADC [3].	21
2-10	ADC under (a) foreground calibration (b) background calibration	22
2-11	Two different schemes for detection and correction of timing errors.	23
2-12	Background timing error detection using reference lane ADC	24
3-1	Error signal computation using a reference lane for timing error estimation.	26

3-2	Estimation of input signal derivative by using additional reference lane with delayed input.	27
3-3	Block diagram of timing calibration loop using two reference lane-ADCs. . . .	28
3-4	Linearized model for timing error calibration loop	29
3-5	Full (0 to f_S) signal spectra for input, output without calibration and output with calibration	30
3-6	Convergence of timing error calibration loop for different input signal statistics. ($\mu_t = 1/(2e7)$)	31
3-7	Gain mismatch between two reference lanes.	31
3-8	Effect of REF-1 lane ADC's input offset on the loop convergence.	32
3-9	Effect of adding high-pass filter on the convergence of the calibration loop. . .	33
3-10	Convergence of calibration loop for different values of x_{IN} , f_{IN} and μ_t	34
3-11	Effect of finite quantization of the two reference lanes on the loop convergence.	35
3-12	Reducing static error by adding uncorrelated dither to the input of the two reference lanes.	36
4-1	Loading of input buffer by single ADC and a time-interleaved ADC alongwith its reference lane.	38
4-2	RC models for sampling front end of one out M-interleaved ADCs in the absence and presence of reference lane.	38
4-3	Two phases of operation for time-interleaved ADC:a) Calibration phase and b) Normal phase of operation.	40
4-4	Estimation of input signal derivative by using additional reference lane with delayed input.	40
4-5	Simulation of interaction between main-lane and reference lane sampling. . .	41
4-6	Effect of C_{REF} on the interaction between main-lane and reference lane sampling.	42
4-7	Using wire based delay lines to isolate interaction between main-lane and reference lane sampling instances.	43
4-8	Impact of mismatch between dummy lane and reference lane, and its dependence on the size of the sampling capacitor.	45
4-9	Validating eqn.(4-13) and eqn.(4-20) for spread in C_1	47
4-10	Minimizing timing error due to mismatch between reference lane and dummy lane by scaling the sampling capacitor.	48
5-1	System-level block diagram of timing error calibration loop.	52
5-2	Block diagram of clock path.	54
5-3	Sampling edge tuning circuit and its simulation results.	55
5-4	Simulation results of the track-and-hold circuit for main-lane and 16x scaled reference lane.	57
5-5	Simulation results of the track-and-hold circuit for main-lane and 16x scaled reference lane.	58
5-6	Design of HOLD buffer for REF lane ADCs.	60
5-7	Simulation Results of HOLD buffer.	62

5-8	Window of sampling instance due to various sources of timing errors.	63
5-9	Fine-coarse arrangement and unit cell of the DAC.	63
5-10	Circuit-level implementation of 11-bit DAC.	65
5-11	DNL simulations of DNL of 11-bit DAC.	68
5-12	Area estimation of 11-bit DAC.	69
A-1	Signal power level expected at the output of the wideband capture ADC. . . .	76
C-1	Track-and-hold circuit used in this thesis.	82
C-2	Magnitude and phase plot for output of the track-and-hold circuit.	83

List of Tables

5-1	Device sizes for the track-and-hold circuit.	59
5-2	Device sizes for the boot-strap circuit	59
5-3	Specifications achieved by the HOLD buffer design.	61
5-4	Specifications achieved by the 11-bit DAC design.	69
A-1	Summary of specifications for wideband capture ADC for mobile applications	76

Acknowledgment

Last two years that I have spent in Netherlands were a collection of bittersweet memories, both at TU Delft and Broadcom. I take this opportunity to thank all those who cheered with me during my "ups" and stood besides me during my "down" turns.

First and foremost, I would like to express my sincere gratitude to my mentor at Broadcom, Dr. Frank van der Goes, for his continuous support and encouragement. I am indebted to him for his faith in me, being tolerant to my weaknesses and sharing his immense knowledge of circuit design. There were innumerable instances when he allowed me to explore different fields while pulling my focus back on actual problem. I profusely thank Dr. Klaas Bult for giving me an opportunity to work at Broadcom, Netherlands, among an outstanding group of engineers. His intuitive and "equation-free" insight, creativity, and practical bent of mind have greatly enriched my learning and also benefited this work. I had received excellent guidance and had enlightening discussions with both Frank and Klaas. It was indeed an honor to work with both of them. They have sparked many research ideas which I will continue to pursue in future.

I would also like to extend my thanks to Prof. Michiel Pertijs and Prof. Leo de Vreede for reviewing this thesis and agreeing to serve on my thesis defense committee.

My heartiest thanks and appreciation go out to Prof. Kofi Makinwa for financial support, mentorship, test-chip support, and for reviewing my paper and this thesis. Through his actions he has set an example as a mentor and a researcher, which I can only aspire to come close. On the whole, I had a quite exciting experience working under him and I hope to continue interacting with him in the future. I also wish to thank Prof. J. Huijsing for his great enthusiastic support that I received during a hobby project. Prof. Huijsing has a remarkable ability to decompose a complex circuit into its most fundamental form. I am infinitely grateful to him for teaching me finer nuances of analog circuits and allowing me to attend the famous opamp design course.

Last one year at Broadcom has been a remarkable and fulfilling experience that has shaped my thoughts and will steer my future actions. During my initial days, our wonderful HR Els van Zijl, ensured that the cumbersome official formalities are kept smooth and simple. I am truly indebted to her for earnest efforts and help while I moved to a new house. Special thanks are due to Dr. Jan

Westra for keeping our work environment stress-free with his witty sense of humor. I also thank him for proof reading this thesis. I would also like to single out our receptionist, Cora Gunsing for being a "One Stop Solution" for all practical matters. I thank Dr. Davide Vecchi and Rob de Haas for helping in a hundred different ways. Thanks are also due a plenty to all my colleagues, the Chinese group, the Italian gang, and all others, for being such a good company.

I was very fortunate to be a part of coveted Electronics Instrumentation (EI) lab. I would like to thank all my friends and colleagues there for making it an incredible place to work. In particular, I thank Saleh and Ugur for proof reading my thesis, and Zu-Yao for his technical support. Thanks are due to rest all with whom I had hours of fun and fascinating discussions and who have shared many precious moments with me. I owe special thanks to Ugur for his help and support to meet pressing tape-out deadline. Special gratitude goes to the EI Lab office staff, Joyce and Karen, for all the help from managing finance and supporting conference travel to organizing day-out and new-year dinners.

I would like to thank Mirita for her loving support and encouragement. Her positive attitude and never give-up spirit has been an important driving force to see me through the tough times.

Finally, and most importantly, I would like to express gratitude to my parents, my brother and my grandfather. This work is a product of their endless love and support.

Chapter 1

Introduction

INTEGRATED circuits technology has undergone rapid scaling since its invention. In 1960, Gordon Moore made an empirical observation that the number of devices on a chip approximately doubles every 18 months. With scaling to new technology nodes, feature size shrinks by 70%, transistor density doubles, wafer cost increases by 20% and chip cost comes down by 40%. These new technology nodes result in faster, smaller and cheaper transistors. In fact, due to scaling the price of one transistor has dropped 100 million times and the trend still continues. This rapid drop in price of a transistor has stimulated new applications and enabled advanced communication/computing appliances. By making the transistors and the interconnects smaller, more circuits can be fabricated on each silicon wafer thereby reducing the price of each circuit. Additionally, circuits of various nature like digital, analog, and RF circuits can be integrated together to reduce the price of the total system.

1-1 Motivation

Wireless systems like mobile phones are one of the many systems that have benefited greatly from the integrated circuit technology. For instance, integrated radio subsystems for GSM [4], WLAN [5], and Bluetooth [6] have already been demonstrated in the literature. Present mobile-handsets contain many such transceivers of different standards, packed densely together. On the other hand, new wireless standards are being rapidly introduced in the market. The consumer demand of accessing these new wireless services from a single handset is thus, ever increasing. In the rush of providing more wireless services, future mobile handsets will only grow in size and cost which would eventually defeat the benefits derived from technology scaling [7]. This proliferation in size and cost of mobile-handsets can be subdued by integrating various wireless services using a universal and tunable hardware platform. Such a platform should be able to tune to a carrier frequency over a wide range, and should support a variety of modulation and data rates.

One such possible architecture, as proposed in [1], is shown in fig.(1-1). The receive link (Rx) which comprises of an amplifier and an ADC would receive RF signals from all wireless standards. Based on the service required by the user, the digital signal processor (DSP) will tune respective carrier frequency and demodulate the received data. Similarly, depending on desired wireless service, the data

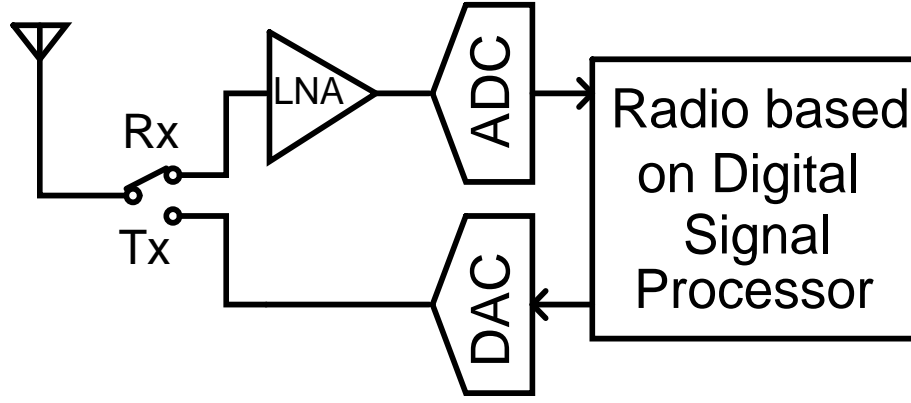


Figure 1-1: Mitola's wideband radio [1]

is accordingly modulated by the DSP which is transmitted directly by a DAC. Thus, all functionality of a radio are implemented on a highly tunable platform of DSP. Finally, in addition of being tunable, a fully-digital radio implementation (like on a DSP) is also favored by steady performance improvement of digital circuits with technology scaling.

The focus of this research is to improve performance of the Rx ADC for wideband radios targeted for mobile applications. As the ADC is close to the antenna, it should support finer input signal amplitude and wider input frequency range. This translates into a stringent requirement on linearity, speed and resolution of the ADC. For instance, a wideband receiver for cable applications is designed to simultaneously capture 16 channels located between 48-1002MHz TV band, with 50dB SNDR (signal-to-noise and distortion ratio) 2.6GS/s sampling speed, and 10b quantization while consuming 0.5W power [8]. Note that a higher resolution may also be required in order to support complex modulation schemes across various wireless standards.

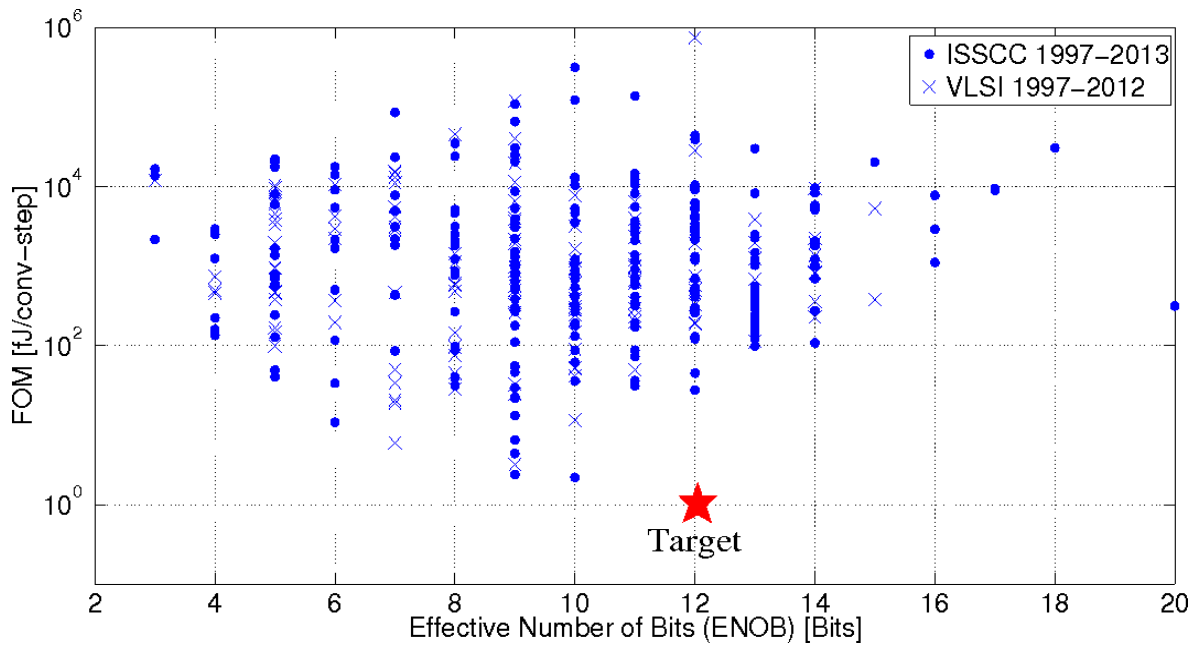
The accuracy of an ADC can be expressed by effective number of bits (ENOB) which is defined as,

$$ENOB = \frac{SNDR - 1.76}{6.02} \quad (1-1)$$

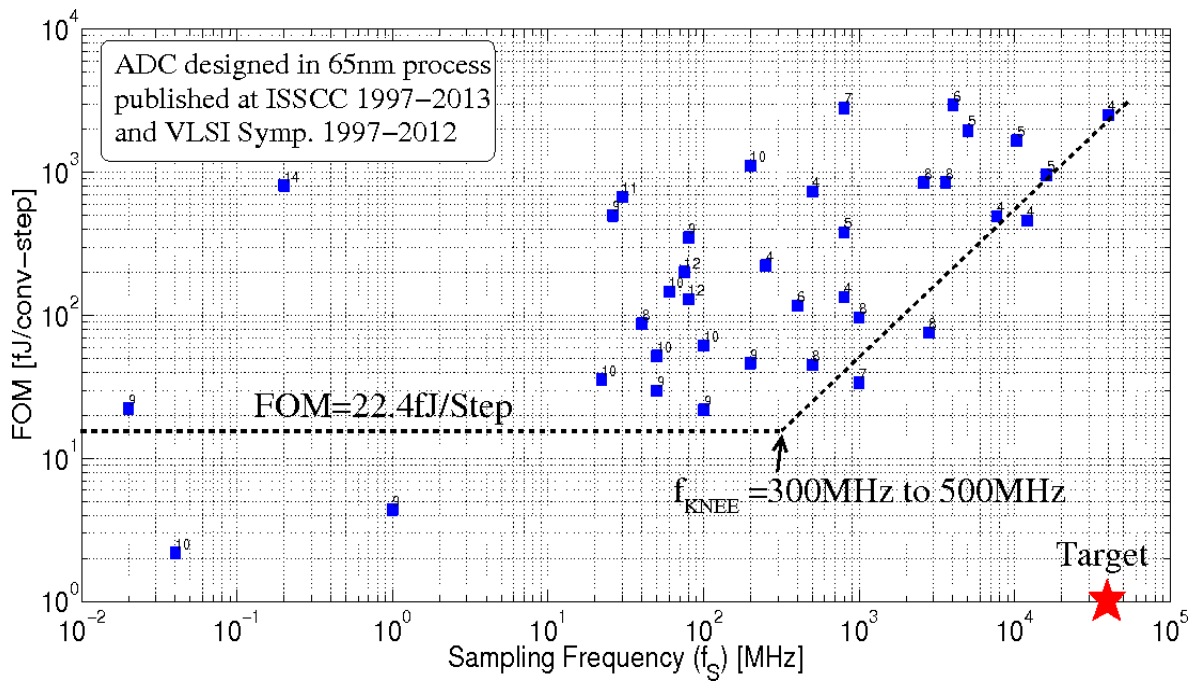
where, $SNDR$ is defined as ratio of input signal power to total noise and distortion power. Depending on the design an ADC can consume varying amount of energy to achieve same ENOB. In order to compare energy efficiency of two different ADC design, a popular figure-of-merit (FoM) used in the literature [2] is given as,

$$FoM = \frac{P_{Total}}{2 \cdot \min(f_S/2, ERBW) \cdot 2^{ENOB}} \quad (1-2)$$

where, P_{Total} is the total power consumed by the ADC, f_S is the sampling frequency, and $ERBW$ is the effective resolution bandwidth. Fig.(1-2)(a) is the plot of figure of merit (FoM) versus ENOB, for all ADCs published at ISSCC and VLSI symposium between 1997 and 2013. A larger value of FoM means that the ADC consumes more energy to while it makes one step to convert analog input to digital value. It is evident that FoM for ENOB of 12 bits is in the order of 20fJ/conv-step which is far from target FoM of 1fJ/conv-step (derived in Appendix-A). The FoM increases rapidly for a higher value of ENOB. FoM is also high for lower ENOB because, these designs primarily have high sampling speed designs. It is evident from the plot that best state-of-the art FoM lies between



(a) FoM versus ENOB.



(b) FoM versus sampling frequency for ADCs designed in 65nm CMOS process only. Number on each data point is the ENOB of that design.

Figure 1-2: FoM versus ENOB and sampling frequency for ADCs published at ISSCC and VLSI from 1997-2013 [2]

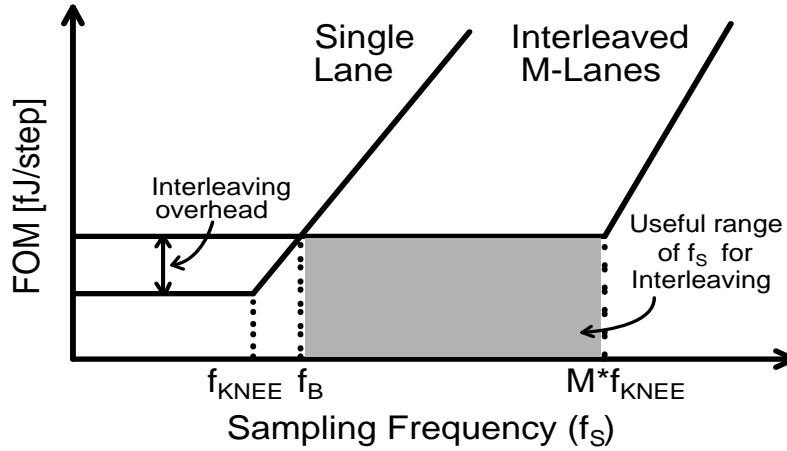


Figure 1-3: FoM versus sampling frequency for single lane and M-lane interleaved ADC

an ENOB of 8 to 9. A similar observation can also be made for FoM versus sampling frequency. The plot in fig.(1-2(b)) shows FoM versus sampling frequency for ADCs designed in 65nm CMOS process. The number on each data point represents ENOB of the design calculated using eqn.(1-1). For sampling frequencies below the knee frequency f_{KNEE} , the FoM curve is flat as the power consumption (P_{Total}) increases linearly with f_s . But for f_s greater than f_{KNEE} , the speed-power relation turns non-linear and hence, mediocre improvement in f_s is achieved at considerable power penalty. The value of f_{KNEE} is determined by technology. For instance, f_{KNEE} is between 300-500MHz for CMOS 65nm process. Note that the FoM plot in fig.(1-2(b)) is on log-log scale and hence, for f_s greater than f_{KNEE} , the FoM increases linearly.

As shown in fig.(1-3), interleaving can break this speed-power trade-off and can improve the FoM of ADC. The technology limit, f_{KNEE} , for a single channel can be extended to $M \cdot f_{KNEE}$ by interleaving M-lanes operating at f_{KNEE} . Due to overhead associated with interleaving, the benefits in FoM are not immediately visible. There exist a break even frequency f_B above FoM improves by interleaving. In other words, an interleaved ADC is more efficient than a single ADC for f_s above f_B . For f_s below f_B a single ADC is more efficient than an M-interleaved ADC because the hardware required to implement interleaving contributes only to P_{Total} in eqn.(1-2). On the other hand, for f_s greater than $M \cdot f_{KNEE}$ the power consumption increases more rapidly for M-interleaved ADC than that for a single ADC. Hence, M-lane interleaved ADC would improve FoM only for sampling frequencies greater than f_B and smaller than $M \cdot f_{KNEE}$.

1-2 Basics of Time-Interleaved ADC

In a time-interleaved ADC many slow ADCs operate in parallel to achieve higher net throughput. The concept of time-interleaved ADCs is not new and was proposed first in [9]. However, the critical advantage of time-interleaved ADC is that it can break speed-power trade-off in a given technology. This benefit was not recognized till a decade ago. The research of [10] is mostly one of the first work to show that for a given speed and resolution overall power consumption of an ADC can be reduced by operating multiple slow single ADCs in time-interleaved fashion. Since then research on time-interleaved ADCs has been quite active.

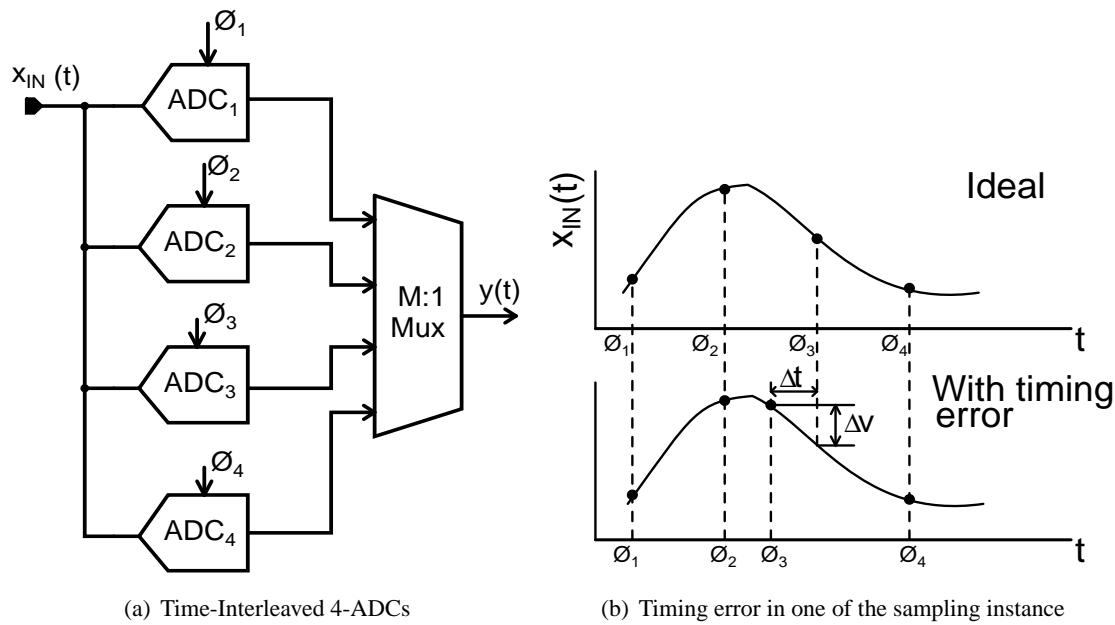


Figure 1-4: Block diagram of basic 4-ADC interleaved with timing error.

Fig.(1-4)(a) shows a simple time-interleaved ADC where 4 individual ADCs are operating in parallel. During ϕ_i the input is sampled by the respective ADC_i . Under ideal conditions the overall output $y(t)$ is obtained by adding the output of all 4 ADCs together. However, in real time-interleaved ADCs there are interleaving issues like input offset, gain mismatch, timing error and bandwidth mismatch. These issues are dealt in greater detail in Chapter-2. For sake of argument let's assume that only timing error, Δt , is present (sources of timing error are covered in Chapter-2). As shown fig.(1-4)(b), due to the timing error the sampling instance of ADC_3 is shifted by Δt . This makes ADC_3 sample input signal at a wrong time instance making an error of Δv . Now, when the outputs are added together, along with the input signal information an additional spurious tone is also created due to the error voltage Δv .

The focus of this thesis is to minimize Δt by employing a calibration loop around the time-interleaved ADCs. As will be shown in the subsequent chapters, by suppressing Δt as much as possible, the magnitude of the spurious tone can be reduced. Spurious tones are also generated by other interleaving issues. However, the problem of input offset and gain mismatch can be solved like in any other ADC but the problem of a timing error is very specific to time-interleaved ADCs. Even though significant amount of work on timing errors has already been done, it is still an active area of research in current literature as it is not yet completely solved. Finally, out of all the interleaving issues timing error is the most difficult to calibrate because it does not easily lend itself to detection or correction which makes the design of calibration loop all the more interesting. Hence, the focus of this thesis is to solve timing errors out of all the other interleaving issues.

1-3 Application: Wideband Capture ADC

Time-interleaving is a well-suited architecture to realize wideband capture ADCs for mobile applications as it can achieve high sampling-speed and low power consumption simultaneously. To better

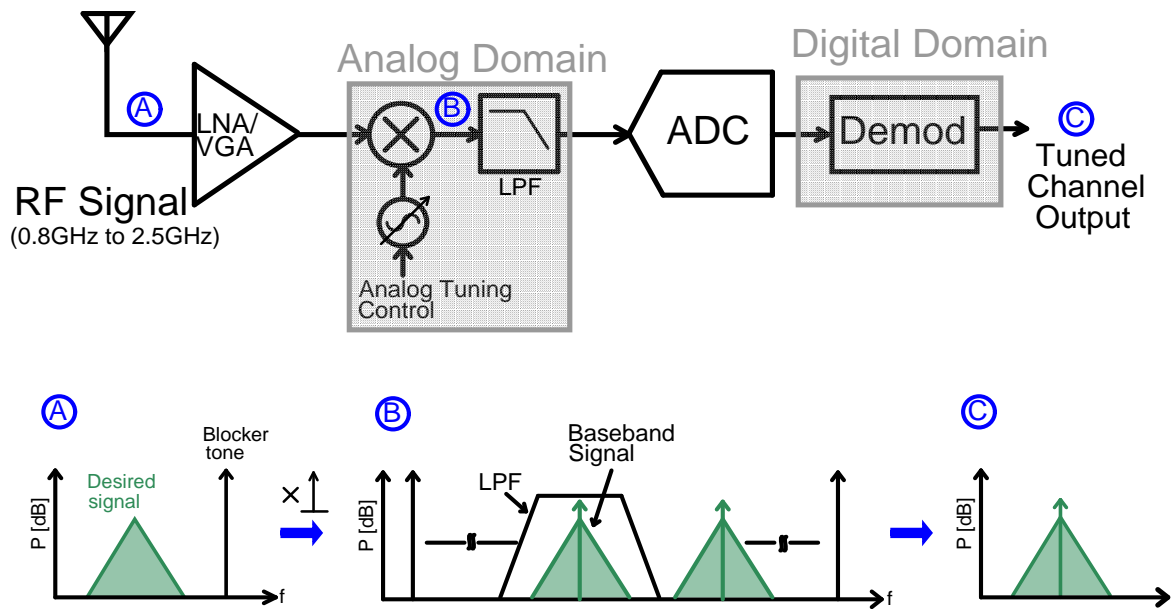


Figure 1-5: Typical Rx link for zero-IF receiver.

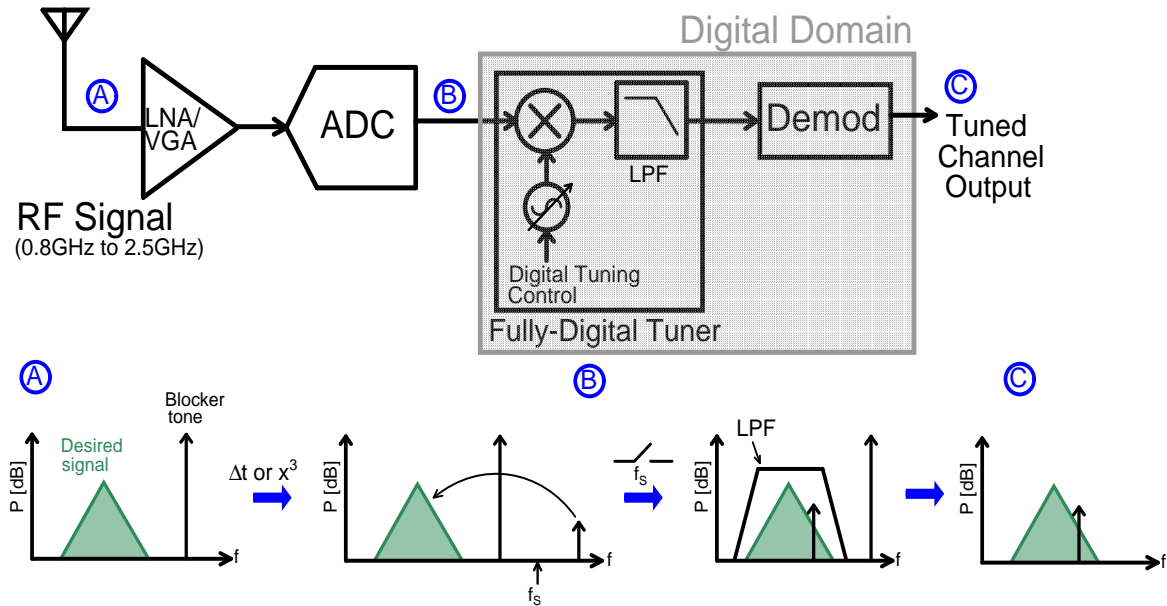


Figure 1-6: Wideband capture receiver architecture for future mobile applications.

understand the functioning of a wideband capture ADC, it is worthwhile to understand the role of an ADC in conventional Rx link. Fig.(1-5) shows one such Rx link for zero-IF receivers. An antenna would receive an input signal from 0.8GHz to 2.5GHz, which covers all the major bands used by today's mobile handsets [7]. Typical signal spectrum at the input of the LNA is shown by (A) in fig.(1-5) where the desired signal is received along with a blocker tone. If the LNA is assumed ideal then this spectrum will appear at the input of mixer unscathed. The mixer down-converts the received signal by multiplying it with local oscillator frequency. Thus, as apparent from (B), the desired signal is moved to baseband and can be retrieved by simple low-pass filtering. The receiver can tune to a new channel by analog tuning of the local oscillation frequency (shown in fig.(1-5)). Thus, as depicted from the spectrum at (C), the presence of a blocker tone does not hamper the receiver functionality.

Fully digital radio of fig.(1-1) can be realized by swapping the position of mixer and low-pass filter with that of the ADC. Thus, as shown in fig.(1-6), mixer and low-pass filter are moved completely into digital domain which relaxes their design. For instance, channels can be easily tuned by changing the digital tuning control word. As argued in the first section, this flexibility enables the implementation of a radio on a DSP platform. However, as the ADC moves closer to the antenna, its design specifications get more stringent and thus, the ADC design becomes more involved. For further apprehension, assume again that the spectrum of the received signal is represented as shown by (A) in fig.(1-6). Due to the timing error, Δt of the time-interleaved ADC an image is created for the blocker tone. The image is only created for the blocker tone and not for the desired signal as the desired signal has much smaller amplitude level. Further, due to the sampling process of the ADC which samples at the rate of f_S , this image of the blocker tone can fold-back into the desired signal band as shown in (B) in fig.(1-6). Once the image of blocker tone lands in the desired signal band even digital filtering will not help. Hence, as shown by the spectrum at (C), the image of the tone will persist at the output as it is indistinguishable from the desired signal.

Hence, designing a wideband capture receiver relaxes design specifications on the mixer, filter and channel selection, but it greatly increases the design challenge for the ADC. As the ADC will be invariably a time-interleaved ADC, its timing error should be low enough to avoid creating strong image of blocker tones.

1-4 Target Timing Error Correction Accuracy

While designing the calibration loop to compensate for the timing errors, it is important to know the level of accuracy expected from such a calibration loop. The wideband capture ADC would capture all wireless standards present in a mobile handset, including GSM cellular. As per GSM specification the sensitivity of the receiver should be at least -100dBm. The range of received signal is from -10dBm to -100dBm. Consider a case as shown in fig.(1-7), where a blocker tone has power of -10dBm whereas, the desired signal is at the lowest supported power level, i.e. -100dBm. Due to timing error and sampling process of an interleaved-ADC, an image of blocker tone might be created in the desired channel bandwidth. Clearly, in order to distinguish the desired signal the image of the blocker tone should be lower by at least 90dB or more. Thus, the timing error should be low enough such that the image of blocker if created, stays below 90dB.

The relation between the timing error and the SNR is given as (derived in Chapter-2),

$$SNR = \frac{3}{2\pi^2 \Delta t^2 f_{IN}^2} \quad (1-3)$$

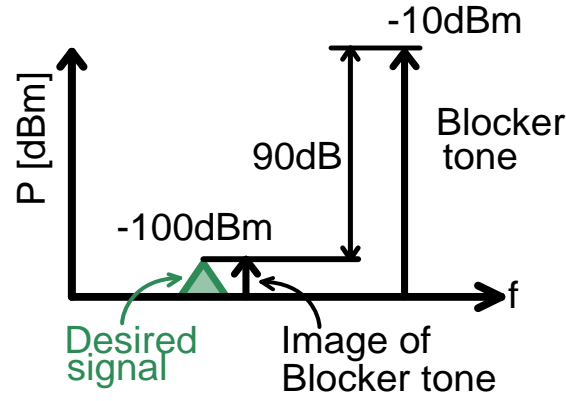


Figure 1-7: Spurious free dynamic range requirement for wideband capture ADC for GSM signals.

For $\text{SNR}=90\text{dB}$, $f_{IN} = 2.5\text{GHz}$, the resulting timing error is 5fs. Thus, the timing correction accuracy of the calibration loop should be at least 5fs or better (Refer Appendix-A for additional specifications). To achieve this accuracy is the target of this thesis.

1-5 Research Goal and Contributions

As argued in the previous section, the sampling time errors create unwanted spurious tones which cannot be distinguished from the desired signal. These spurious tones can be suppressed by calibrating the sampling time error. In order to receive a GSM signal the sampling time errors of the wideband capture ADC should be below 5fs level. To achieve this level of accuracy a calibration loop is needed. Hence, the initial goal of this research is to investigate various calibration architectures and choose the one which can achieve timing error correction accuracy of better than 5fs.

For achieving high timing error correction accuracy, it is vital to detect timing errors precisely. In this regards, it will be shown in Chapter-2 that the use of additional reference ADCs, is the most suitable topology [11]. From the results presented in [11] it is not evident that this topology can truly achieve 5fs accuracy. However, they do reveal that this topology is quite promising. Thus, the second goal of this research is to study this calibration architecture, identify any existing issues, and provide relevant solutions to overcome them.

As it will be shown in chapters-3 and 4, the topology of [11] has several shortcomings. For instance, the input offset of the reference lanes hampers the calibration loop convergence. Also, the use of reference lanes to detect timing error changes the load of the input buffer whenever it is used by the calibration loop. This change in the loading of the input buffer limits the accuracy with which the timing errors can be detected. Lastly, the sampling instance of reference lanes and the main-lane ADCs interact with each other to further degrade the timing error detection.

Finally, even if the above mentioned problems are solved, the calibration loop proposed in [11] can achieve 5fs accuracy only when a high resolution D/A converter is available to correct the timing errors. This D/A converter needs to have low-area and low-power as multiple copies of it would be employed.

The key research contributions which addresses all these goals are highlighted below:

- The calibration loop is made resilient to the input offset of the reference lane by incorporating a high-pass filter in the loop.
- The change in input buffer loading due to reference lanes, is solved by adding dummy lanes. Thus, making the loading of input buffer constant.
- The interaction between the sampling instance of the reference lanes and the main-lane ADCs is identified through analytical derivation. It is also cross verified through simulations. This problem is solved by inserting delay lines in the sampling front-end of reference lanes and main-lane ADCs.
- For correcting the timing errors a low-power 11-bit D/A converter is designed which achieves DNL of $< 1LSB$. The D/A converter has sufficient dynamic range to correct errors within $\pm 2.5ps$.

1-6 Thesis Organization

This thesis is divided into six chapters. Chapter-2 begins with an overview of time-interleaved ADCs and provides a brief explanation of various time-interleaving issues on the performance of an ADC. As the focus of this thesis is to calibrate timing errors, more emphasis is laid on it. Sources of timing errors and its impact on signal-to-noise ratio (SNR) are explained. Lastly, calibration topologies and techniques to detect timing errors are discussed.

In chapter-3, a background timing error calibration loop is discussed which uses two reference lanes for error detection and adjusts sampling clock edges for timing error correction. The stability and convergence of the calibration loop are studied using a MATLAB model. Stability of the loop is examined in the presence of input offset, gain mismatch, and varying input signal amplitude and frequency. Further, it was observed that finite quantization of reference lanes can cause an error in the steady-state convergence of the loop. This problem is solved by adding dithering at the input of the reference lanes. The effectiveness of adding dither is studied and relevant simulation results are discussed in this chapter.

Chapter-4 investigates the timing errors caused due to loading of the reference lane. Reference lanes only load the input buffer during calibration phase and not during the normal ADC operation. This modulation of input buffer load also contributes to a timing error. To avoid this problem, the reference lanes are replaced by a dummy sampling front-end during normal operation mode. Further, during calibrate phase the sampling instances of reference lane and main-lane can interact with each other adding to the timing errors. To mitigate this interaction, a solution of using delay lines in the sampling circuit is proposed. Lastly, mismatch between a dummy lane and a reference lane also contributes to timing error. A simple technique is proposed where scaled-down version of the main-lane sampling circuit is used for reference and dummy lanes to reduce the timing errors.

Chapter-5 shows circuit level implementation details of various blocks used in the timing error calibration loop. Design of blocks such as sampling front-end circuit of main-lane ADC and reference lanes, D/A converter for tuning sampling clock edges, and the reference lane buffer, are described along with circuit simulation results.

Finally, chapter-6 draws a conclusion from this work and potential topics for future improvements and research are suggested.

Chapter 2

Sampling Time Errors in Time-Interleaved A/D Converters

The time-interleaved ADC cycles through M-parallel lanes of ADCs generating a net throughput M-times higher than a single individual lane ADC. As a result, a time-interleaved ADC can achieve high sampling speeds which would not be possible with a single ADC without excessive power penalty. Thus, for a given resolution, time-interleaved ADCs can break the speed-power trade-off by reducing total power consumption and increasing the sampling speed. However, mismatch between the lane-ADCs degrade the performance of the total interleaved-ADC by introducing unwanted spurious tones in its output spectrum. Thus, it counteracts the speed benefit gained from time-interleaving.

In this chapter, the basic principles of time-interleaved ADCs is described. It is followed by a brief discussion on various sources of mismatch like offset, gain, timing and bandwidth. The focus of this thesis is mainly on calibration of timing errors. Hence, the topics pertaining to timing errors like their effect on ADCs performance, methods to detection and calibrate them, and calibration topology are elaborated.

2-1 Basics of Time-Interleaving

In a typical time-interleaved ADC, M-number of ADCs operate in parallel with a conversion rate of f_S/M , where M is the number of interleaved ADCs and f_S is the sampling rate. These M-interleaved ADCs operate from individual sampling phases, ϕ_i (where, $i = 1, 2, \dots, M$), which are phase shifted from each other by one sampling period T_S . During these sampling phases, each individual ADCs sample the input signal. At the output, data from all M-interleaved ADCs is muxed together to achieve an overall sampling rate of f_S .

To further understand the operation of the time-interleaved ADCs, consider an ideal 2x interleaved ADC as shown in fig.(2-1). The input signal, $x_{IN}(t)$ is sampled by two ideal ADCs, ADC_1 and ADC_2 . These two ADCs operate with phases ϕ_1 and ϕ_2 , and generate output data represented by $y_1(t)$ and $y_2(t)$ respectively. For a hypothetical input $x_{IN}(t)$, the output of two ADCs, $y_1(t)$ and $y_2(t)$

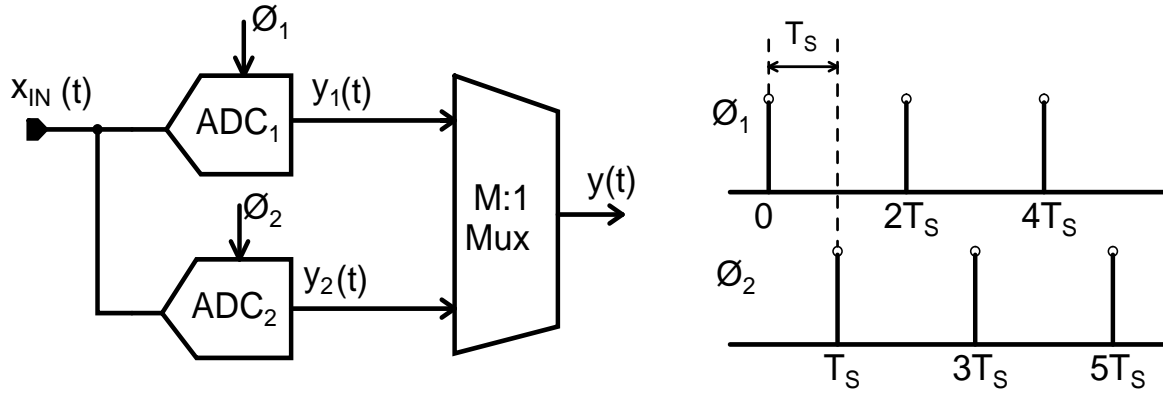


Figure 2-1: Block Diagram of an ideal 2x interleaved ADC with timing waveforms.

can be pictorially presented as shown in fig.(2-2). It would be apparent that both ADCs operate with their individual lane sampling rate of $f_S/2$. Thus, if $f_{IN} > f_S/4$ then both $y_1(t)$ and $y_2(t)$ contain aliases of the input signal. However, when $y_1(t)$ and $y_2(t)$ are added together the final output contains no aliases. This phenomenon is depicted in frequency-domain picture of the interleaved ADC as shown in fig.(2-2).

The reason behind the absence of aliases is that, as shown in fig.(2-1) the sampling instance of each individual ADCs are exactly one T_S apart. Thus, ϕ_2 is time-shifted by one sampling period with respect to ϕ_1 . In frequency domain this time-shift corresponds to the rotation of phase. As the shift in time is exactly one clock period, the phase of the aliases in the output of ADC_2 are exactly 180° out of phase with respect to the aliases in the output of ADC_1 . Fig.(2-2) shows the spectra of the input signal and the outputs of the two ADCs. As the aliases in $Y_2(f)$ are 180° out of phase with that in $Y_1(f)$, when added together they cancel each other exactly. An interesting mathematical account of this alias cancellation is exhaustively derived in [12].

Thus, M-interleaved ADCs operating at f_S/M , under ideal conditions, generates output identical to an individual ADC operating at f_S . Unfortunately, due to device mismatch, each lane-ADCs have slightly different offset, gain, bandwidth and sampling time instances. Due to these non-idealities, alias images in the output of each lane-ADC do not exactly cancel and thus, leave behind some residual alias images. Depending on the source of non-ideality, these residual alias images create spurs in the overall output spectrum.

2-2 Types of Interleaving Issues

In this section various interleaving issues are discussed with more emphasis being laid on timing errors. A basic MATLAB script used to simulate these errors is provided in Appendix-B.

2-2-1 Offset Mismatch

Figure.(2-3(a)) shows a typical 4-interleaved ADC with corresponding input offset. This offset arises from the input pair of a comparator used in an ADC and it is a random but additive error. In absence

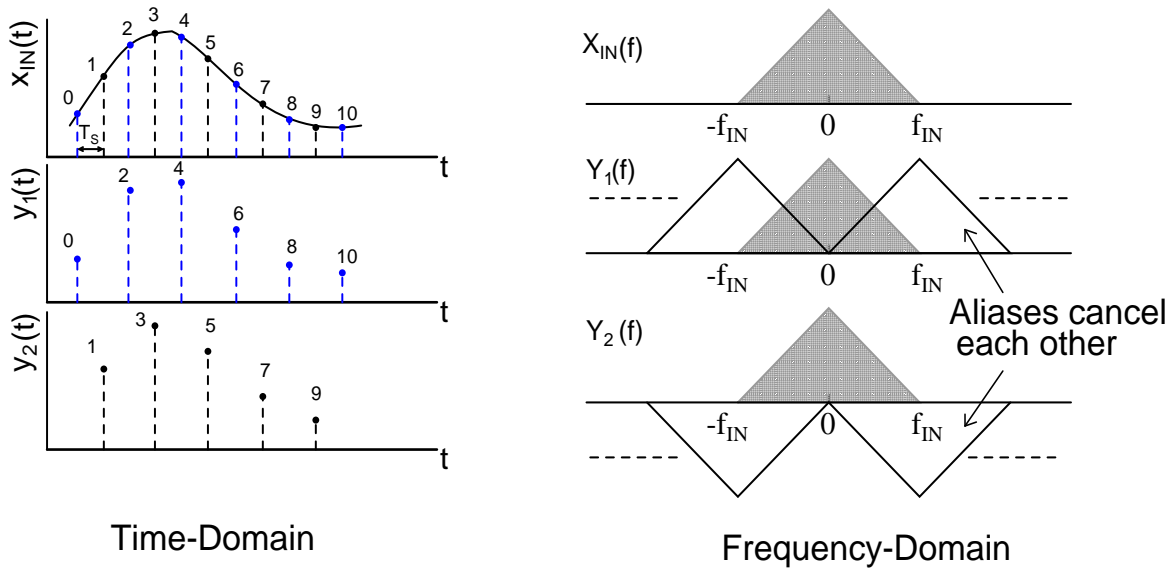


Figure 2-2: Input and output signal spectrum of ideal 2-interleaved ADCs

of input signal, each lane-ADCs will sample their own offset which when combined with the output of another lane-ADCs would generate a periodic error signal. This periodic error signal creates spurious tones which are located at [3],

$$f_{SPUR,OS} = k \cdot \frac{f_S}{M} \quad \text{where, } k=1,2,\dots,M \quad (2-1)$$

A typical spectrum of the input signal and the lane-ADC output is as shown in fig.(2-3(b)). The arrows (blue) are the tones due to sampling of input offset. These offset tones manifest themselves as spurs in the overall output spectrum of ADC as shown in fig.(2-3(c)). The time domain error signal due to different input offset of the lane-ADCs can be seen in fig.(2-3(d)). The magnitude of the offset spurs depends upon the amplitude and shape of this periodic error signal. However, these spurs are independent of input signal frequency and amplitude. Thus, they can be removed by employing techniques like digital filtering [13], chopping [14], or even calibration [11].

2-2-2 Gain Mismatch

As shown in fig.(2-4(a)), assume that all 4-lane ADCs have different gains (A_1 to A_4). All the other characteristics are perfectly identical and ideal (e.g. zero input offset). Gain error is defined as the maximum difference between the gain of any two lane ADCs. The primary sources of gain error is the difference in reference voltages or the differences in the sampling circuit (e.g. charge injection or clock feedthrough) between various lane ADCs. Figure.(2-4(b)) shows that the gain error manifests itself by modulating the amplitude of the output spectrum. Thus, when the output of all the lane ADCs is combined, the alias in the output spectrum do not necessarily cancel each other. The magnitude of these spurs due to residual alias, depends on the magnitude of gain error and input amplitude.

Similar to offset mismatch, gain error also creates a periodic signal with frequency f_S/M . For a sinusoidal input, this periodic signal creates spurs by mixing with the input signal. These spurs,

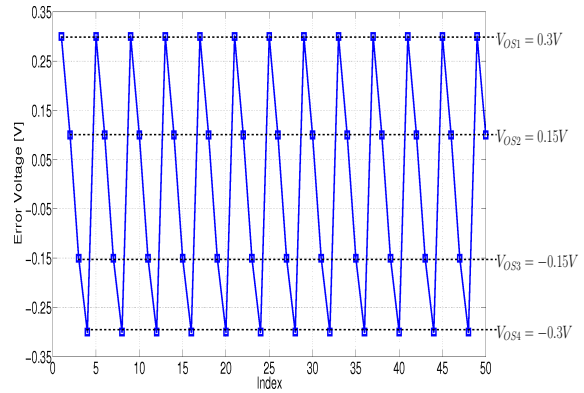
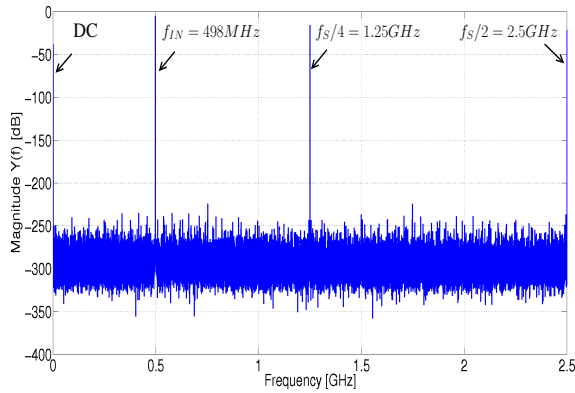
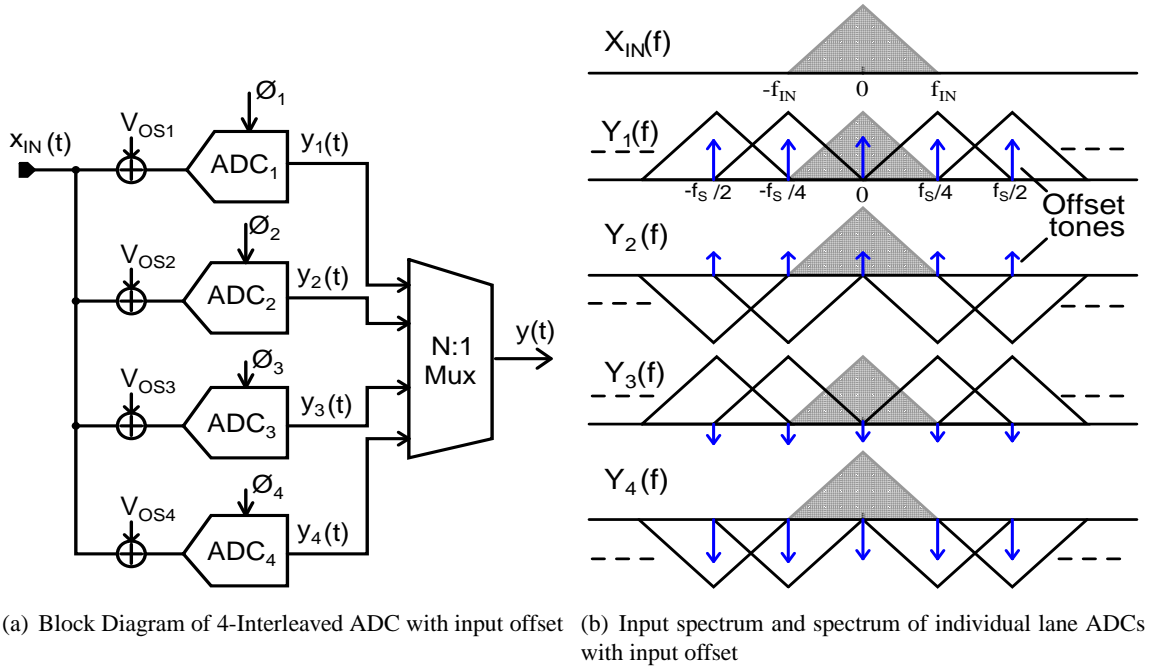
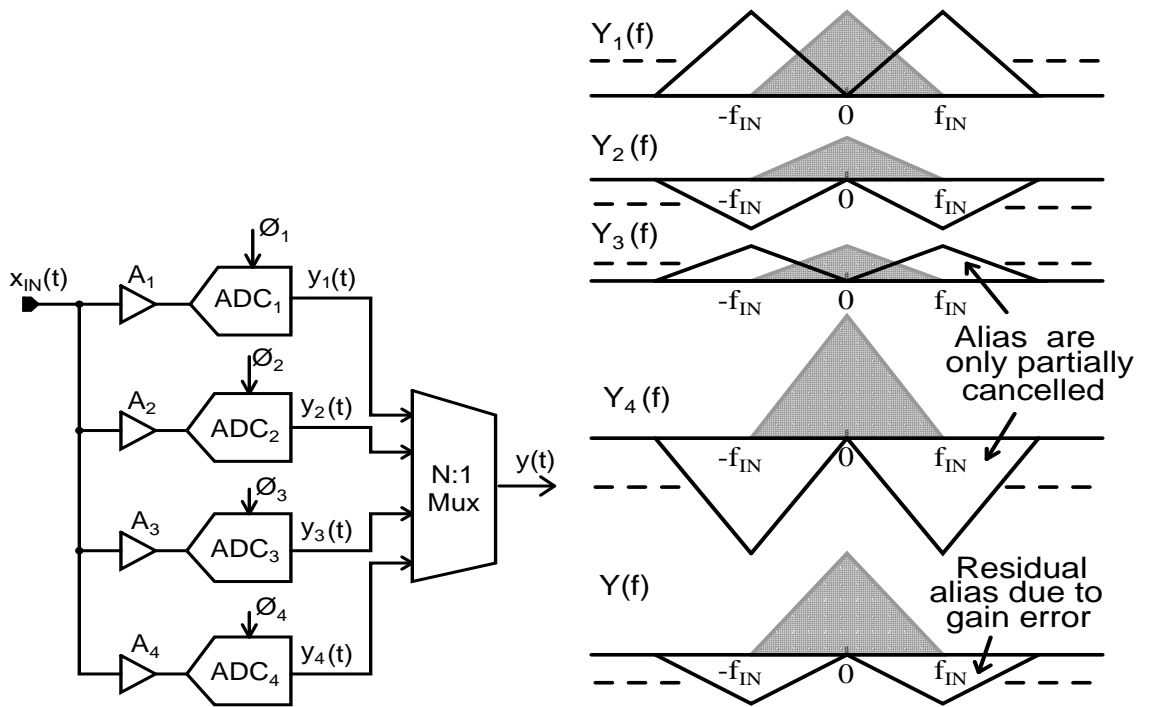
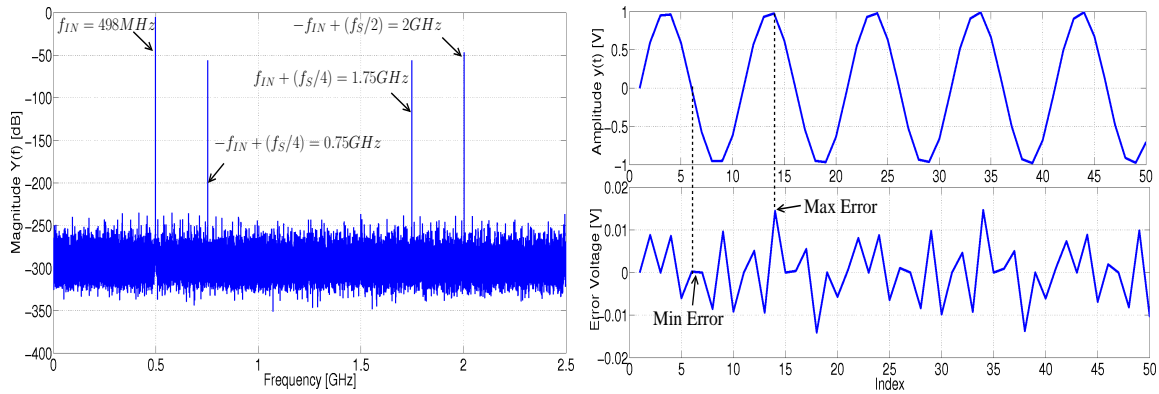


Figure 2-3: Effect of input offset and its mismatch on the output spectrum of 4-interleaved ADCs for sinusoidal input tone of $f_{IN} = 498MHz$ sampled at $f_s = 5GHz$



(a) Block Diagram of 4-Interleaved ADC with gain error

(b) Spectrum of individual lane-ADCs and total output with gain error

(c) Output spectrum showing spurs due to gain mismatch of $-40dB$ (d) Error signal for input sinusoid of $f_{IN} = 498MHz$ sampled at $f_S = 5GHz$ **Figure 2-4:** Spurs due to gain error in the output of a 4x interleaved ADCs

$f_{SPUR,GE}$, are located at [3],

$$f_{SPUR,GE} = \pm f_{IN} + k \cdot \frac{f_S}{M} \quad \text{where, } k=1,2,\dots,M \quad (2-2)$$

The location of these spurs can be verified from fig.(2-4(c)). Note that with a change in the input frequency, only the location of $f_{SPUR,GE}$ changes and not its magnitude.

The magnitude of the gain error spurs is modulated by the input signal amplitude. Thus, the largest error in the output would occur when the input amplitude is the largest. As shown in fig.(2-4(d)), for a sinusoidal input, the maximum error in the output occurs at the peak of input amplitude and it is minimum when input crosses zero. Thus, like amplitude modulation, the spurs due to gain error are also multiplicative in time domain.

Unlike, the offset mismatch correction, the correction of gain mismatch is slightly involved. The gain mismatch can be easily calibrated by employing a foreground calibration which is not always possible. Hence, at present background calibration of gain mismatch is also a quite active research area. Research of [15], [16] and [17] are some of the popular works that can be used to combat the gain mismatch related errors.

2-2-3 Timing Mismatch

Ideally the phase difference between the adjacent lane clocks, which are separated by one sampling period T_S , should be equal to $2\pi/M$. In practical implementation, the phase or timing errors are unavoidable due to the finite propagation of the clock signal, and variations in the clock buffers and sampling switches. For high input signal frequencies even a small timing mismatch can create significant error. The input signal effectively is phase modulated by a periodic timing error signal which has a frequency of f_S/M .

To highlight the impact of sampling time error, consider an ideal 2x interleaved ADC as shown in fig.(2-1). As shown in fig.(2-5), t_1 and t_2 are ideal sampling instances of ADC_1 and ADC_2 which are separated exactly by one sampling period (T_S) under ideal conditions. However, in the presence of a sampling time error, Δt , the sampling instances will shift from their ideal positions. As shown in fig.(2-5) the sampling instance of ADC_2 has shifted from t_2 to $t_2 + \Delta t$. This shift in the sampling instance will create an error voltage Δv_{TE} which can be expressed as,

$$\Delta v_{TE} = \frac{dx_{IN}}{dt} \cdot \Delta t \quad (2-3)$$

Thus, taking this error voltage into account the output of two ADCs at the sampling instances t_1 and $t_2 + \Delta t$ can be written as,

$$y_1(t_1) = x_{IN}(t_1) \quad (2-4)$$

$$y_2(t_2 + \Delta t) = x_{IN}(t_2) + \Delta v_{TE} = x_{IN}(t_2) + \frac{dx_{IN}(t_2)}{dt} \cdot \Delta t \quad (2-5)$$

It can be observed that $y_2(t_2)$ contains the input signal $x_{IN}(t_2)$ which is corrupted by its derivative. The operation of derivative in time-domain corresponds to a phase-shift of 90° in the frequency domain. Thus, the timing error tone will be 90° out of phase with the input signal. Thus, when the

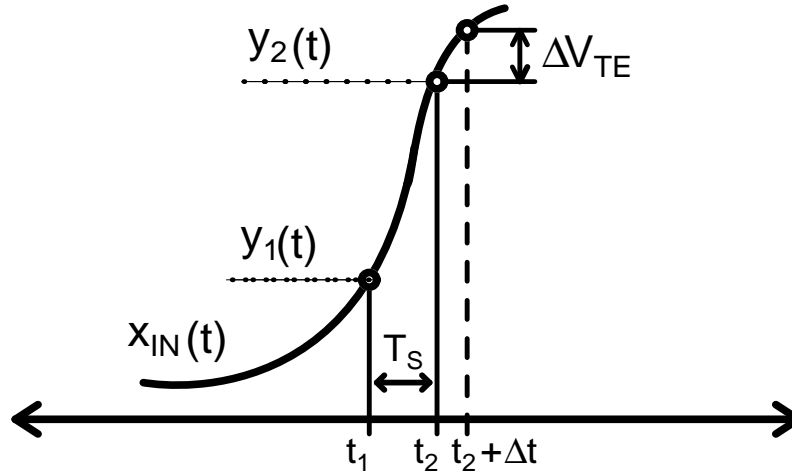


Figure 2-5: Error in sampling input signal due to timing error

outputs of all the lanes are added together, the aliases again do not cancel each other. This creates spurs due to timing mismatch which are located at [3],

$$f_{SPUR,TE} = \pm f_{IN} + k \cdot \frac{f_S}{M} \text{ where, } k=1,2,\dots,M \quad (2-6)$$

The location of $f_{SPUR,TE}$ is same as the location of the spurs that stem from the gain mismatches. Unlike the spurs due to gain mismatch, the magnitude of $f_{SPUR,TE}$ will not only depend upon input amplitude but also on the input frequency. This additional information might be exploited to separate gain error and timing error from each other. For DC input signals, timing error is zero and spurs in the output spectrum are only due to gain error. Once the gain error is calibrated or removed then, for high input frequencies, the spurs left in the output spectrum are due to timing mismatch or bandwidth mismatch.

2-2-3-1 Impact of Timing Error

Assume a noise-free time-interleaved ADC. The output spectrum of such an ideal ADC will still have spurs due to interleaving issues (e.g. offset, gain, timing and bandwidth). Further, assume that these spurs are only due to timing error and not due to offset, gain or bandwidth mismatch. In such a scenario, the magnitude of these spurs will limit the maximum achievable SNR of a time-interleaved ADC. In order to get an estimate of this limit, assume a uniformly distributed input signal with the spectrum as shown in fig.(2-6). A uniformly distributed signal closely resembles a broadband signal which is the expected input signal for a wideband ADC in mobile applications.

On the other hand, the spectrum of the timing error signal, $(2\pi f \Delta t)$, is parabolic and increases infinitely with frequency. This spectrum when multiplied with input signal spectrum creates the total timing error spectrum, as shown in fig.(2-6), that corrupts the output spectrum of a time-interleaved ADC. Hence, it degrades the maximum SNR that can be achieved. To estimate the impact of timing error on the SNR, the input signal power and noise power needs to be computed. The signal power,

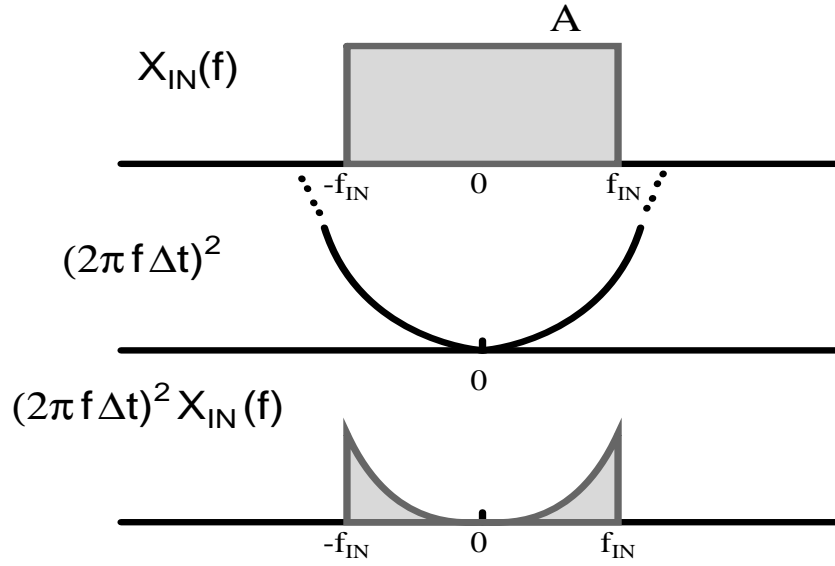


Figure 2-6: Spectra of input signal and timing error component.

P_S , can be calculated by integrating the input spectrum shown in (fig.(2-6)), from $-f_{IN}$ to f_{IN} as,

$$\begin{aligned} P_S &= 2 \int_{-f_{IN}}^{+f_{IN}} A df \\ &= 4 \cdot f_{IN} \cdot A \end{aligned} \quad (2-7)$$

Similarly, noise power can be calculated as,

$$\begin{aligned} P_n &= \int_{-f_{IN}}^{+f_{IN}} (2\pi f \Delta t)^2 X_{IN}(f) df \\ &= 8\pi^2 \cdot \Delta t^2 \cdot A \left[\frac{f_{IN}^3}{3} \right] \end{aligned} \quad (2-8)$$

Solving the eqn.(2-7) and eqn.(2-8) yields signal-to-noise ratio (SNR) as,

$$SNR = \frac{P_S}{P_n} = \frac{3}{2\pi^2 \Delta t^2 f_{IN}^2} \quad (2-9)$$

This expression is an upper bound on the achievable SNR for a given timing error and given input signal frequency. A more elaborated discussion on plausibility of eqn.(2-9) can be found in [12]. Fig.(2-7) shows that for 12-bit required resolution, the timing error for a uniformly distributed input signal, has to be less than 80fs for an input frequency of 500MHz and less than 20fs for an input frequency of 2GHz. Recall that, as argued in Chapter-1 the target specification on the timing error is 5fs RMS.

2-2-3-2 Sources of Timing Mismatch

The timing mismatch arises due to factors like finite skew of clock distribution network, systematic layout mismatches, delay variation of clock buffers, mismatch in the sampling switches, etc. However,

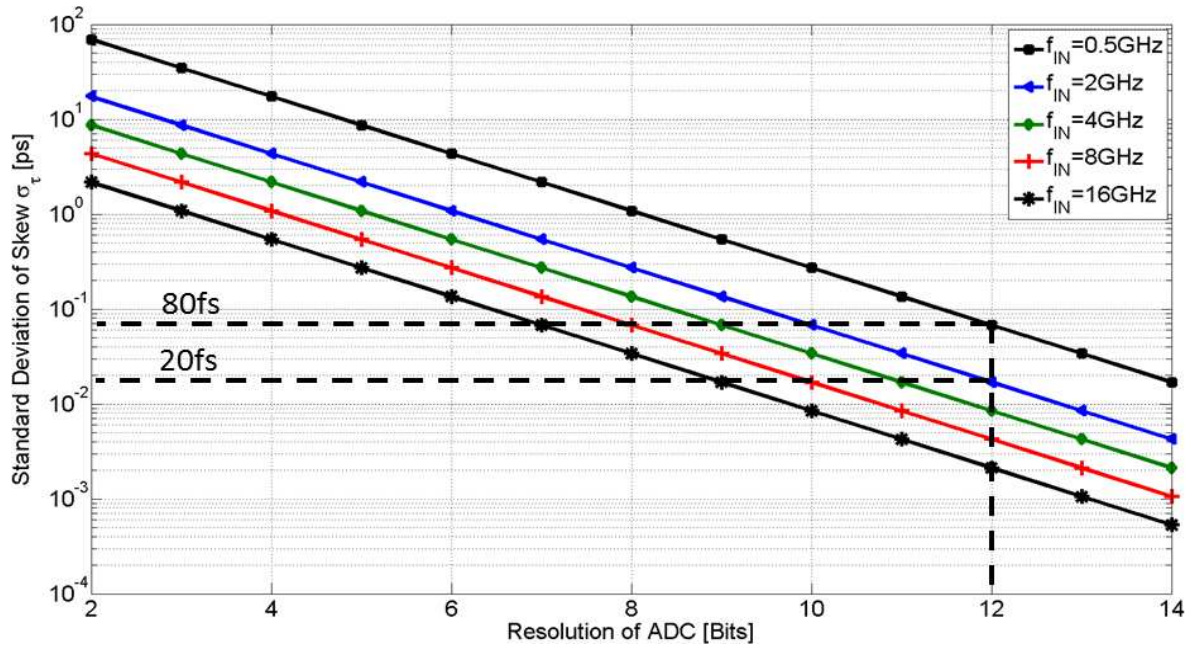


Figure 2-7: Bound on timing error for varying SNR and input frequencies

exact sources of timing mismatch, to certain extent, depend on the specific implementation of the sampling circuit. In this sub-section, the sources of timing error for a fully differential passive track-and-hold circuit, shown in fig.(2-8), are discussed. The design details of this circuit will be discussed in Chapter-5.

Fig.(2-8) shows two track-and-hold interleaved together. It also shows the clock distribution network and clock buffers. During the track phase (ϕ_{Track1} is high) the top-plate switches (M_{T1+} and M_{T1-}) are closed and the voltage on the sampling capacitor (C_S) follows the input signal. During the hold phase the bottom-plate switches (M_{B1}) opens and the top-plate of sampling capacitor is grounded. The differential output ($V_{O+} - V_{O-}$) of the track-and-hold is thus, defined during the hold phase.

One of the timing error sources is a mismatch in the input distribution network which introduces systematic timing skew. Further, the mismatch in clock generation and distribution network also creates timing errors. These mismatches stem from different drive strength of clock buffers due to process variations, layout mismatches in clock buffers, HOLD period (T_{A1} and T_{A2} in fig.(2-8)) mismatch, and timing skew due to mismatches in the clock distribution network (routing of ϕ_{Track1} , ϕ_{Track2} , ϕ_{Hold1} and ϕ_{Hold2}). For a larger interleaving factor, both clock distribution and input distribution become more complex and hence prone to larger mismatches.

In addition, mismatch in the passive sampling circuit also causes a timing error. These includes, mismatch in threshold voltage of bottom-plate sampling switches (M_{B1} and M_{B2}), variation in ON-resistance of top-plate switches (M_{1+} , M_{1-} , M_{2+} , and M_{2-}), and mismatch in sampling capacitor C_S due to process variation. Also, mismatch in bootstrap circuit (represented by C_{BS} and two switches in fig.(2-8)) will also introduce significant timing error.

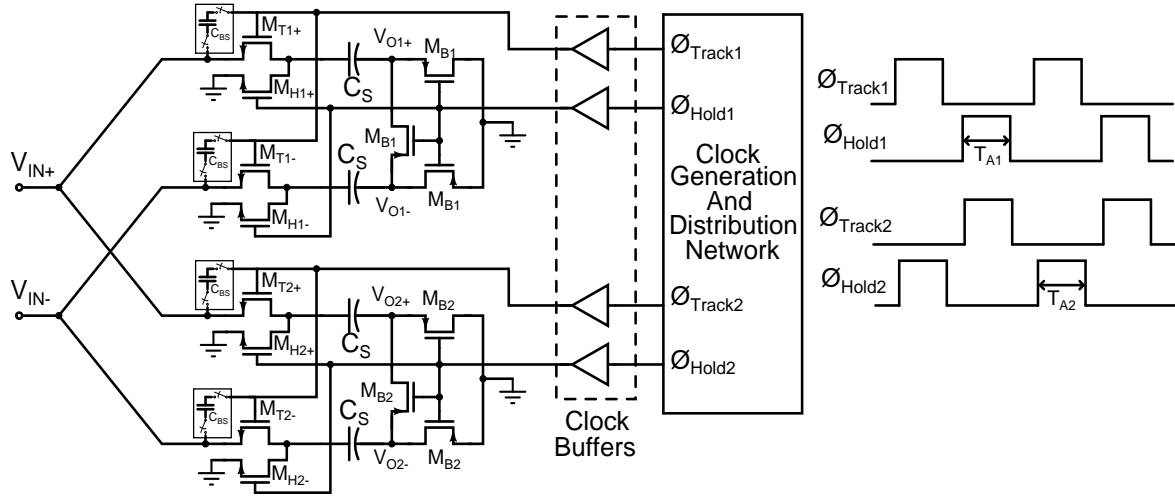


Figure 2-8: Typical track-and-hold circuit for 2x interleaved ADC

2-2-4 Bandwidth Mismatch

The last nonideality that hampers performance of time-interleaved ADCs is the mismatch in bandwidth of the sampling front end. The magnitude and phase response of sampling circuit changes with input frequency due to its finite bandwidth. The mismatch in bandwidth of different lane ADCs, thus, creates gain and timing error which changes with input frequency. To assess the effect of spread in bandwidth ($\sigma(\Delta BW/BW)$) on the magnitude of gain and timing error, the sampling front-end can be modeled as a simple first order RC filter. Detailed derivation and analysis of timing and gain errors using this model can be found in [3].

Fig.(2-9) shows degradation in SNR with increasing bandwidth mismatch $\sigma(\Delta BW/BW)$, for different ratios of input frequency to bandwidth. To achieve SNR corresponding to 12 bits with the sampling front-end bandwidth at least 10x greater than the maximum input frequency, the maximum allowable $\sigma(\Delta BW/BW)$ is about 0.3%. However, if $\sigma(\Delta BW/BW)$ is 0.3% but the bandwidth is just 2x larger than the maximum input frequency then, the SNR degrades down to 9 bits. Thus, to avoid considerable gain and timing errors the bandwidth of sampling front-end should be chosen large enough (about 10x). Alternately, bandwidth mismatch can also be calibrated [18].

2-3 Timing Error Detection and Calibration

As discussed in the previous section, to avoid spurs due to timing mismatch, in high speed time-interleaved ADCs, the sampling edges of the individual lane sampling clocks should be set precisely (e.g. with maximum error of 20fs for 12 bits). Due to various timing error sources like finite propagation delay of clock signal distribution, systematic layout mismatches, delay variation of clock buffers, and threshold voltage variation of the sampling switches, it is impossible to achieve this level of accuracy only with careful layout. Hence, some form of calibration or correction is required to solve the errors due to sampling time.

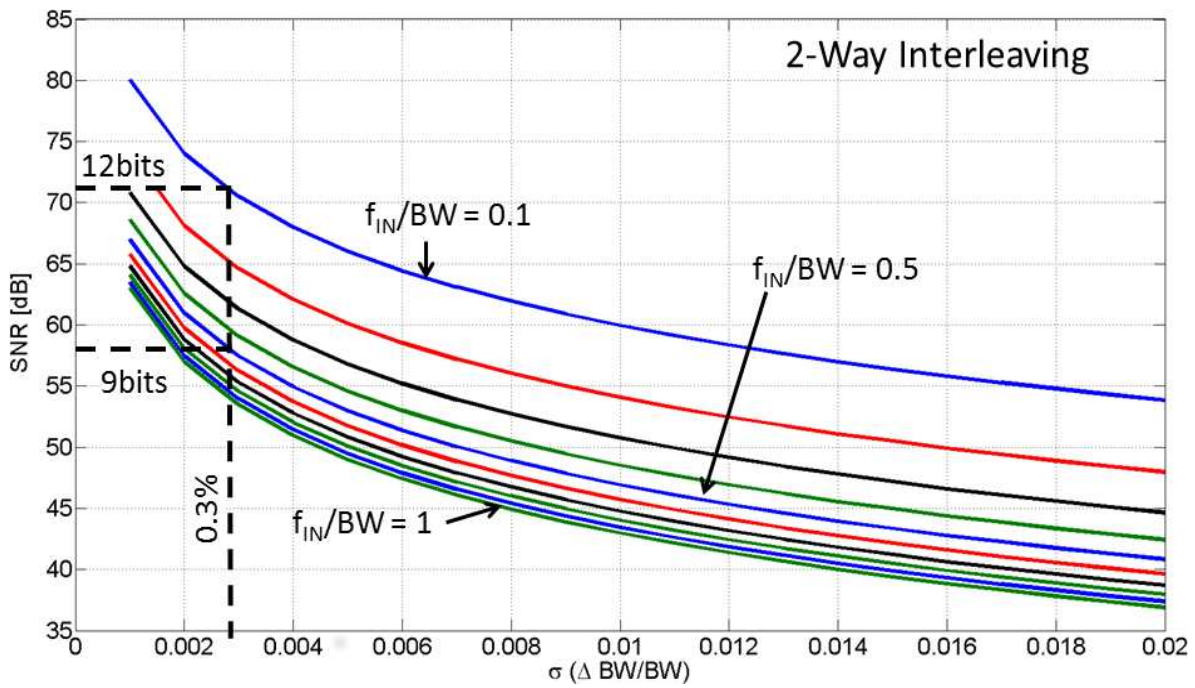


Figure 2-9: Impact of bandwidth mismatch on SNR for a 2x-interleaved ADC [3].

2-3-1 Use of common sample-and-hold.

One straightforward way to mitigate timing errors is to use a sample-and-hold front end common to all the interleaved lanes [8] and [19]. However, the common sampling front-end should work at the sampling rate f_S instead of f_S/M in case of per-lane sampler. The elevated sampling speed either creates harmonic distortion due to limited settling of the input sampler or incurs enormous power penalty as input sampler should operate M-times faster. Further, the noise of input sampler limits the overall resolution achievable by the ADC. Hence, using a common input sampler would be suitable only for moderate sampling speeds and moderate resolution ADCs. In addition, a common input sampler cannot solve bandwidth mismatch unless it is explicitly calibrated or each lane-ADC is ensured to have much wider bandwidth than input signal. Alternately, it would be desirable to part away with a common input sampler and instead have a way to calibrate timing error with lower area and power overhead.

2-3-2 Foreground vs background calibration.

Invariably, all timing-error calibration techniques have two critical components: 1) timing error detection and 2) timing error correction. The detection and correction of timing errors can occur in the foreground or in the background as shown in fig.(2-10). Typically, during foreground calibration a test signal like a ramp or sinusoid is applied and timing error is detected in the digital domain (e.g. by using FFT) [20], [21] and [22]. Timing error correction can be done either in digital domain (e.g. fractional delay filter) or in analog domain (e.g. adjusting sampling clock edges). Such foreground calibration is preferred when the circuit parameters like voltage and temperature or ambient operating conditions do not change dramatically. Foreground calibration is also attractive in applications

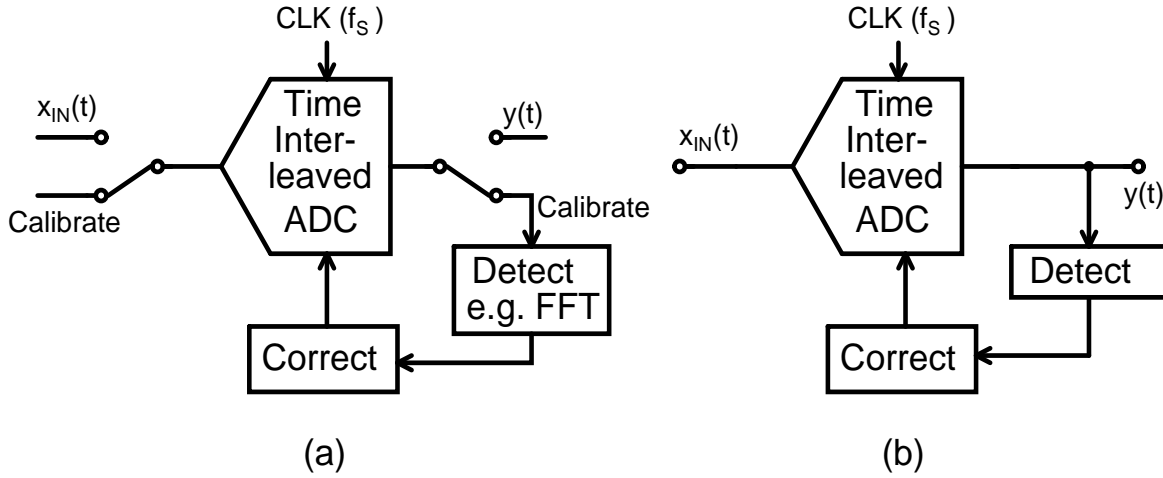


Figure 2-10: ADC under (a) foreground calibration (b) background calibration

like oscilloscopes, where occasionally ADC can be taken offline for calibration. However, for target application (wideband capture ADC for mobile applications discussed in Chapter-1), the ADC cannot be taken offline. Also, the circuit parameters and ambient operating conditions change significantly. In such scenarios, foreground calibration is not feasible and background calibration is the de facto choice. As shown in fig.(2-10)(b) during background calibration, timing error is detected and corrected while the ADC is still continuously operating.

2-3-3 Digital detection and digital correction of timing errors.

Irrespective of the type of calibration, timing error detection and correction techniques can be broadly classified to be either based on digital logic or analog circuitry. Figure.(2-11(a)) shows detection and correction of timing error in the digital domain. The lane ADC outputs, $y_1(t)$ and $y_2(t)$, are passed through an adaptive fractional delay filter whose coefficients are tuned to correct for the timing error. Thus, when the two outputs are added together there is no timing error component in the final output. Based on the detected timing error, the filter coefficient can be tuned using various algorithms. For instance, researchers in [23] and [24] detect timing errors by chopping the output signal and using a 10-bit 21-tap FIR approximation to Hilbert transform filter. To correct the timing error in a 10-bit ADC, they employ a 29-tap FIR filter with 10-bit coefficients that realize an adaptive fractional delay filter. This particular approach has a drawback that if Nyquist-operation is desired then the input signal should be strictly band-limited to $f_s/2$. This condition may not be true for wideband capture ADCs where the spectral content beyond $f_s/2$ may very well have finite energy. Further, the implementation of fig.(2-11(a)) would require the digital logic to observe and correct timing error on per sample basis. Thus, for M-interleaved ADCs, M digital blocks would operate at f_s/M rate. Although this approach might still be tractable for low sampling rates, it would incur very large power penalty for multi-GHz sampling rates. Finally, due to the complex adaptive fractional delay filters, this approach also has increased area and latency penalty.

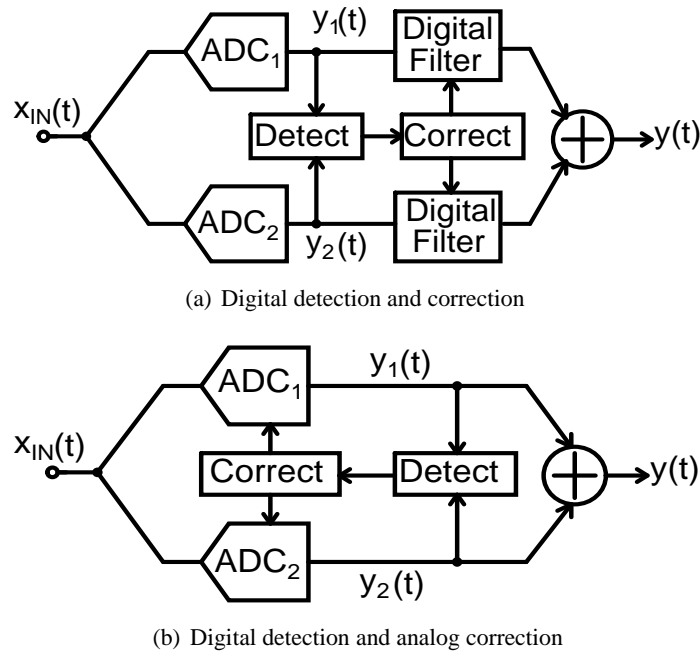


Figure 2-11: Two different schemes for detection and correction of timing errors.

2-3-4 Digital detection and analog correction of timing errors.

Alternately, as shown in fig.(2-11(b)), a second approach would be to detect the timing error in the digital domain and adjust analog circuit to correct for the timing error [20], [21]. As the timing error is compensated and thus physically removed, this technique can be very power efficient. The feedback loop does not operate on every sample, rather it gradually converges to remove timing error. Thus, once the feedback loop converges, it can enter into a low power mode. Analog tuning can be performed by adding a controlled phase shift (all-pass filter) in the input signal path or by controlling the sampling clock edges in the clock path. The latter technique is widely used as adding any analog circuit in the input path would hamper linearity and noise performance of the ADC. The sampling clock edges can be tuned by inserting a variable capacitive load in the clock path or by tuning the drive strength of the clock buffers.

Thus, background calibration of timing error with digital detection and analog correction is best suited for wideband capture ADC for mobile applications. Detection of timing error is challenging especially when the ADC is continuously operating. One solution is to add some form of pilot tone, like a known pseudo random signal or a ramp signal, along with the input. The timing error can then be detected by separating the pilot tone from the input signal by post-processing in background (e.g. low pass filtering [25], [26], and [27]). Unfortunately, invariably all post-processing techniques that can separate input signal and pilot tone rely heavily on the statistics of either input signal or pilot tone or both. For instance, in order to separate the pilot tone (e.g. a ramp signal) from the input signal, the input signal is required to have zero mean. Such constraints on input signal are not always feasible, however it might be possible to impose such constraints on the pilot tone. Still, in order to accommodate the amplitude of pilot tone the dynamic range of input signal has to be restricted. Even if these limitations were to be overcome, the accuracy of detecting timing errors is limited by the poor efficacy of post-processing block to separate pilot tone from input signal.

Alternately, timing error can be detected by using an additional reference lane as shown in fig.(2-12). In [28] a similar approach is used where the background calibration algorithm tries to maximize the correlation between the lane ADC and a 1-bit reference lane ADC. Thus, the timing error is removed by aligning the sampling time instance of all the lane ADCs to that of the same reference lane ADC. In addition, compared with using pilot tone, the timing error detection can be more precise while using a reference lane. This is because when a pilot tone is used the timing error detection accuracy depends on how accurately input signal and pilot tone can be separated, which further depends on the amount of averaging being employed. Whereas, while using a reference lane, timing error is detected from same input sample directly. Hence, in order to meet the specification outlined in Chapter-1, the optimum choice of the timing error calibration loop would be to use background calibration with digital timing detection using reference lanes and analog correction.

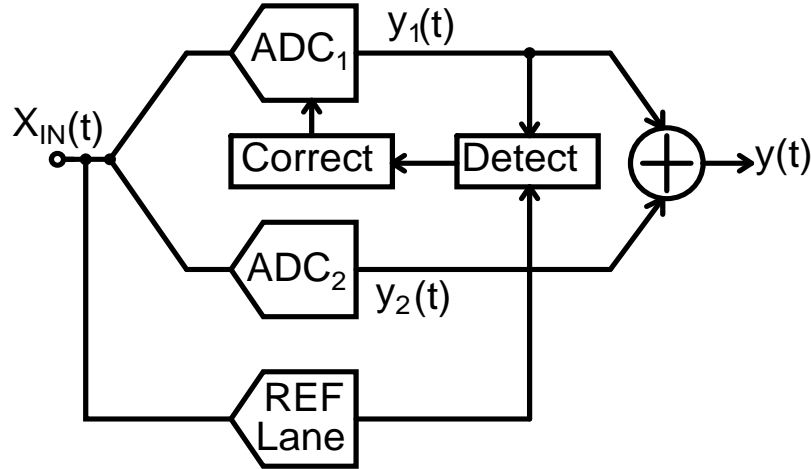


Figure 2-12: Background timing error detection using reference lane ADC

2-4 Summary

In this chapter, the basic operation of time-interleaved ADCs is described. Several non-ideal effects that degrade the performance of such ADC are discussed. However, as the focus of this thesis is to calibrate timing errors in time-interleaved ADCs, they are discussed in greater detail. Formulation of the timing errors, its effect on SNR and its various sources are covered in this chapter. For a 12 bit SNR, the timing error has to be less than 20fs which cannot be achieved merely by careful layout. Hence, various calibration techniques are studied. Looking at the feasibility of calibration and desired precision, it can be concluded that background calibration with timing error detection based on reference lane and correction based on tuning sampling clock edge, is best suited for wideband capture ADCs for mobile applications.

Chapter 3

Timing Error Calibration Loop

SAMPLING time errors should be accurately detected to enable precise correction. As argued in Chapter-2, the timing error information can be separated from the input signal by using reference lanes. As a result, the timing errors can be detected with high accuracy. Based on the detected timing error, the sampling instances of all time-interleaved ADC lanes can be aligned to that of a common reference lane ADC. To fulfill this condition a background calibration algorithm in digital domain can be implemented. Similar attempt is reported in [11]. However, as shown in Chapter-2, for the target application of wideband capture ADC, the timing errors should be corrected with accuracy of 5fs RMS which is not possible if the calibration reported in [11] is used as it is. In this chapter, various limitations of the background timing error calibration loop of [11] is studied. Appropriate solutions are employed to overcome some of these limitations and realize the timing correction accuracy of 5fs. The core of the calibration loop is the use of two additional reference lanes for computing the timing errors. One reference lane quantizes the input signal whereas, the other quantizes the derivative of the input signal. The timing errors are compensated by analog fine-tuning of the sampling clock edges. The factors that govern the stability and convergence of the calibration loop are explored using a MATLAB model. Also, the input offset of the reference lanes or any of the main lanes can hamper the loop convergence. A high-pass filter (HPF) is embedded in the loop to circumvent this problem. Finally, the finite quantization of reference lane can potentially limit the correction accuracy of the loop. This problem is solved by adding dither to the input of the reference lanes. The effectiveness of this solution is studied in this chapter and pertinent simulation results are presented in the sections below.

3-1 Principle of Operation

Assume a simple timing error detection system as shown in fig.(3-1), where an ideal main lane-ADC and an ideal reference lane-ADC sample the input, $x_{IN}(t)$, at the same moment. However, due to various sources of timing error (discussed in Chapter-2) the sampling instance of main lane-ADC is shifted by Δt . The goal of the timing calibration loop is to precisely detect this shift and compensate for it. The output of main lane ADC, $y(t)$, and the output of reference lane ADC, $y_R(t)$, can be written

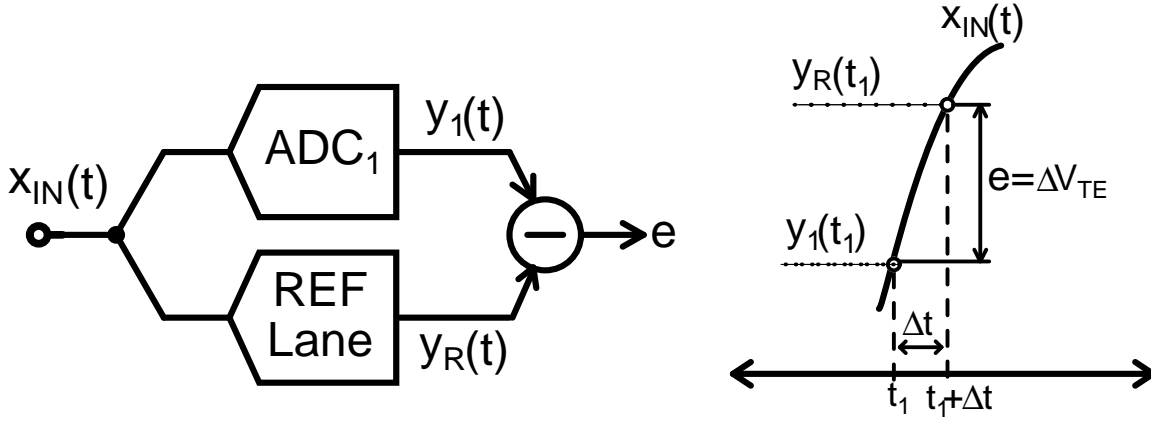


Figure 3-1: Error signal computation using a reference lane for timing error estimation.

as,

$$\begin{aligned} y_1(t) &= x_{IN}(t_1 + \Delta t) = x_{IN}(t_1) + \Delta v_{TE} \\ y_R(t) &= x_{IN}(t_1) \end{aligned} \quad (3-1)$$

where, Δv_{TE} is given by eqn.(2-3) in Chapter-2. The error signal, defined as the difference between the main lane and the reference ADC outputs, can be obtained from eqn.(3-2) as,

$$e = y_1 - y_R = \Delta v_{TE} = \left(\frac{dx_{IN}(t)}{dt} \right) \cdot \Delta t \quad (3-2)$$

$$= D \cdot \Delta t \quad \text{where, } D = \frac{dx_{IN}(t)}{dt} \quad (3-3)$$

Calibration loop can correct Δt if its value is known. However, as shown by eqn.(3-3), the error signal, e , not only depends on Δt but also on the derivative of the input signal. Hence, along with the error signal, it is critical to estimate the derivative of the input signal. One brute force approach would be to compute derivative in the digital domain and use it along with eqn.(3-3) to estimate and correct Δt in one shot. Unfortunately, such digital computation should be very precise which requires considerable amount of hardware at operates at full sampling speed. This will incur heavy area and power penalty. For example, as argued in [29], to keep the error in derivative estimation below half LSB, a 10-bit ADC operating at 3GHz would at least need 20-tap FIR filter with floating-precision coefficients. Alternatively, the precision requirement on the computation of the input signal derivative can be relaxed if a feedback loop is employed which tunes some analog knob (e.g. the sampling clock edges) to minimize Δt . In this case the precision of the input signal derivative should be just good enough to ensure that the loop converges. As shown in fig.(3-2), a derivative of the input signal can be easily estimated by using an additional reference lane whose input is delayed by a fixed fraction t_D . The value of the delay element t_D is not required to be known accurately as long as it stays constant with input amplitude and frequency. However, to keep the timing error detection accuracy intact for the input frequencies close to Nyquist frequency, the maximum input frequency should be much smaller than $1/t_D$. The delay element, t_D , can be implemented either in the input signal path or in the clock path. In the input signal path, it can be implemented by inserting a small series resistor. As the track bandwidth is the product of sampling capacitor and ON resistance of sampling switch,

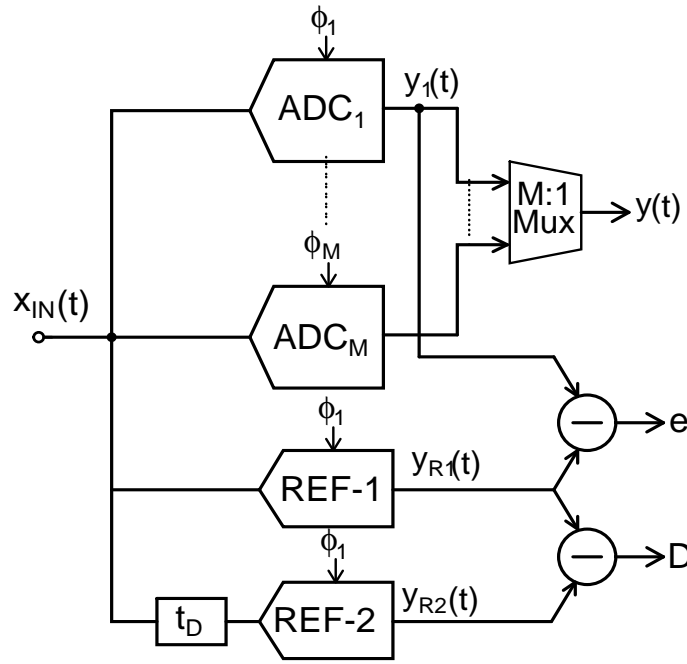


Figure 3-2: Estimation of input signal derivative by using additional reference lane with delayed input.

adding a series resistor will effectively change the track-bandwidth thereby implementing the required delay t_D . Unfortunately, as this series resistor is in the input signal path, it would inevitably contribute to the noise and distortion performance of the ADC. On the other hand, inserting t_D in the clock path would simply mean delaying the sampling clock edges which can be done by adding a capacitive-D/A converter at the output of the clock buffer.

However, sometimes having an additional reference lane for estimating a derivative can pose considerable overhead (e.g. when small interleaving factors are used). Hence, it would be convenient if the derivative can be obtained by using succeeding interleaved lanes whose clock edge is one sampling period delayed. For example, as shown in fig.(3-2), the derivative of the input signal can be obtained by using the output of ADC_M instead of using a dedicated REF-2 lane. Both of these approach would yield exactly the same result if the delay element t_D is chosen to be one sampling period T_S . The difference is that the method of using one of the interleaved lanes would have low estimation error when the input frequencies are much below the Nyquist frequency. For higher input frequency the estimate of the derivative gets much worse. In fact, if the calibration loop uses this method for derivative estimation then it can work only for input frequencies smaller than the Nyquist frequency [29]. Whereas, when the derivative is estimated using an additional REF-2 lane, the criteria to be ensured is that the maximum input frequency is much smaller than $1/t_D$. Thus, the method of using additional REF-2 lane is more robust and accurate. Hence, it is preferred for detecting timing errors in this thesis.

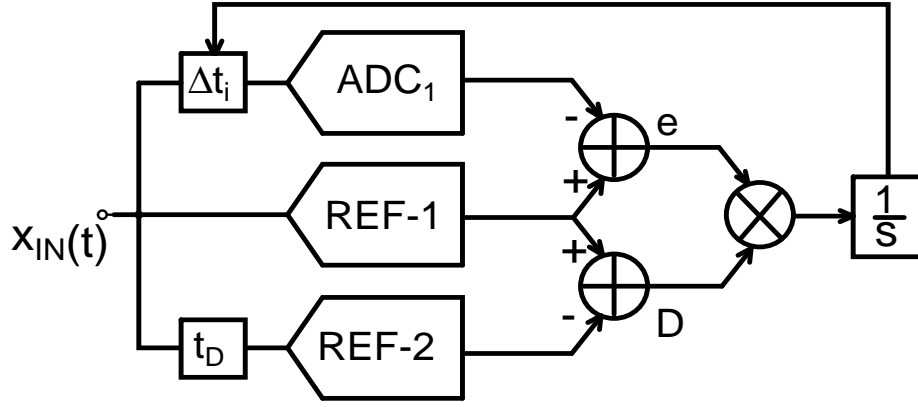


Figure 3-3: Block diagram of timing calibration loop using two reference lane-ADCs.

3-2 Description of Calibration Loop

Fig.(3-3) shows the block diagram of timing calibration loop which uses an additional reference lane to estimate the derivative of the input signal. The timing error is corrected by a feedback signal Δt_i . As this feedback signal is in the analog domain, the accuracy requirement on computing derivative is greatly relaxed. The simplest method to find Δt_i such that it corrects the timing error is by employing an iterative algorithm. Most popular iterative algorithm is the LMS algorithm. By applying LMS equation to eqn.(3-3) the new value of Δt_i can be known [29]. Thus, the value of $\Delta t_{(i+1)}$ is updated based on the following LMS relationship,

$$\Delta t_{(i+1)} = \Delta t_i - \mu_t \frac{de^2}{d(\Delta t)} = \Delta t_i - 2\mu_t e D \quad (3-4)$$

where, μ_t is the LMS coefficient for the timing calibration loop. As shown by eqn.(3-4), to detect the timing error a product of the error signal e and the derivative of the input signal D needs to be computed. This computation is crucial as it removes the dependency of the error signal e from the sign of the input signal. As shown by eqn.(3-3), the error signal not only depends on Δt but also on D . Depending on the slope of the input signal, the sign of D can be either positive or negative. If the feedback loop operates only by observing the error signal e then its direction can change if the slope of input signal flips. Thus, it will be highly unstable. If the feedback loop operates on the LMS equation given by eqn.(3-4) then this problem of stability is resolved. Now, the integrator can control Δt_i through a negative feedback such that the mean of the product $e \cdot D$ is driven towards zero. For a timing mismatch of Δt , if the mean of $e \cdot D$ is negative then the loop would increase Δt_i towards Δt . Whereas, if the mean of $e \cdot D$ is positive then the loop will reduce Δt_i towards Δt . When the loop converges, the mean of $e \cdot D$ is zero. Thus, the loop is completely independent of the slope of the input signal. Depending on the value of μ_t the negative feedback can be made unconditionally stable which is necessary for the calibration loop to converge.

Fig.(3-4), shows a linearized model of the calibration loop. The value of μ_t controls the amplitude of the signal at the input of integrator. Thus, it damps the loop gain and thereby, enhances the stability of the timing calibration loop. If μ_t is sufficiently low then the stability of the loop is increased. However, the update rate, as seen from eqn.(3-4), is reduced considerably which increases the convergence speed of the calibration loop.

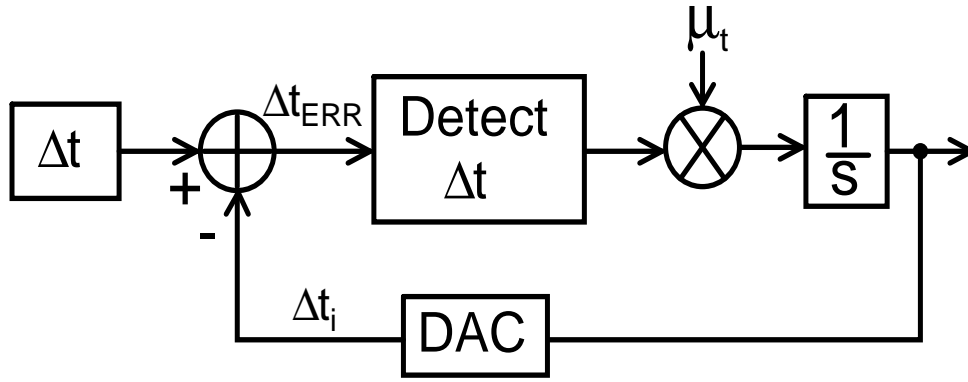


Figure 3-4: Linearized model for timing error calibration loop

The calibration loop functions completely in background which is a primary requisite as the main lane ADCs cannot be taken offline for calibration. However, the reference lanes ADCs lend themselves for foreground calibration which makes the calibration of offset, gain, timing error and bandwidth mismatch between the two reference lanes, much easier.

3-3 Convergence of the Calibration Loop

A non-converging calibration loop would rather exacerbate the timing error problem rather than solving it. Hence, some understanding of the strength and weaknesses of the algorithm needs to be analyzed. Particularly, when the derivative of the input signal is grossly approximated by quantizing its time shifted version using an additional reference lane. Instead of using rigorous analysis, the convergence of timing calibration loop is studied using simulations based on a MATLAB model of the loop shown in fig.(3-3). The feedback signal Δt_i is corrected in steps of 5fs with dynamic range of 3ps. The value of LMS coefficient μ_t was derived through trial-and-error method using MATLAB simulations. During the simulation μ_t of $1/(2e7)$ was used. In this section, the convergence of the loop is studied in the presence of different input signal statistics, input offset, and gain mismatch.

3-3-1 Input Signal Statistics

As shown by eqn.(3-3), the error signal depends on both Δt and the input signal derivative, D . The LMS feedback loop iteratively tries to extract Δt by using estimate of D from an additional reference lane. Under ideal conditions the LMS algorithm will converge exactly to Δt and will suppress any input signal traces present due to D . However, in practice the reference lanes have finite quantization. Also, the value of D is approximated by quantizing the time-shifted version of the input signal. Due to this approximation eqn.(3-4) will have residual traces of the input signal which might deteriorate timing error detection.

To investigate this effect, a simple method is to check the loop convergence in presence of input signals with different statistics like fixed tone or uniformly distributed. Fig.(3-5) shows the input spectrum till f_S for two-tones located at 2.27GHz and 2.47GHz. Assuming a typical timing error, Δt of 1.497ps, it is evident that before calibration the spurs due to timing error are only 45dB below

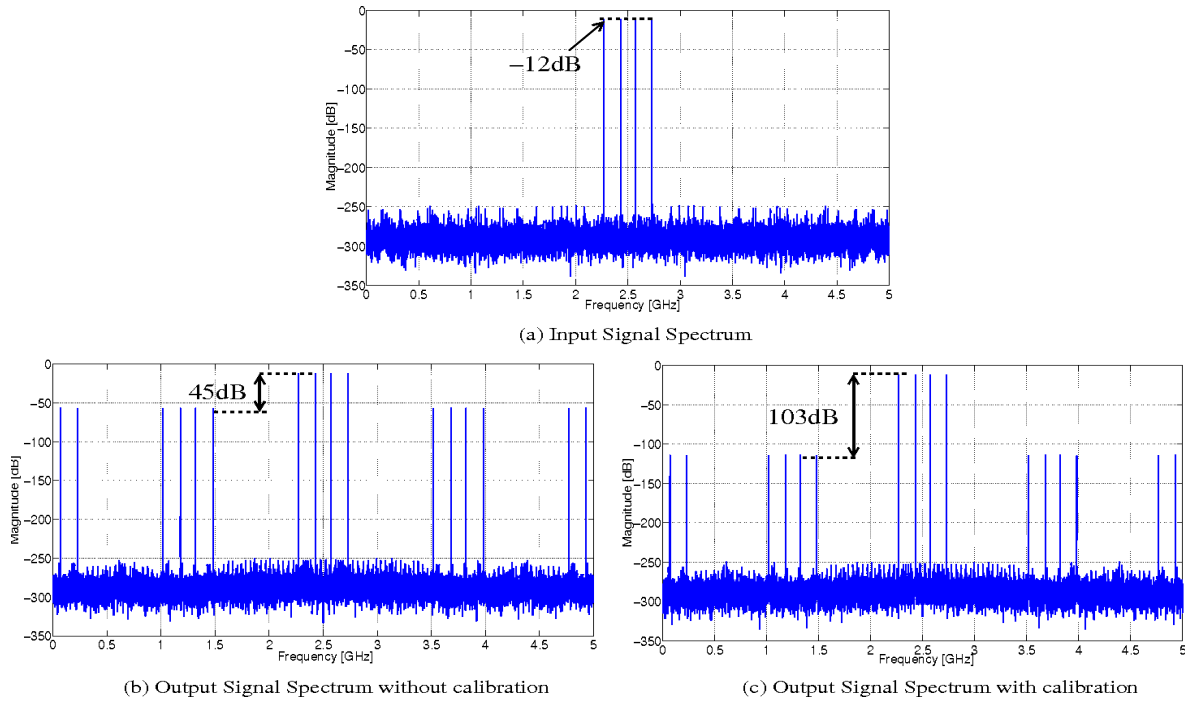


Figure 3-5: Full (0 to f_S) signal spectra for input, output without calibration and output with calibration .

the two-main tones. However, after calibration these spurs are pushed to 103dB level. Fig.(3-6)(a) depicts the behavior of the calibration loop. The feedback signal Δt_i converges to the actual timing error, $\Delta t = 1.497\text{ps}$ within a step-size of $\pm 5\text{fs}$ whereas the error signal Δt_{ERR} converges to zero with $\pm 5\text{fs}$ error. The inset picture confirms that the convergence error is within 5fs. Next, instead of fixed two tones as input signal, a uniformly distributed random signal is used. Fig.(3-6)(b) shows that the calibration loop is able to detect timing error even in the presence of random input signal and converge to correct timing-error ($1.497\text{ps} \pm 5\text{fs}$) value.

3-3-2 Gain mismatch and offset of the two reference lanes

If the two reference lanes have different gains they amplify input signal differently which again manifest as residual input signal in $e \cdot D$. The calibration loop cannot distinguish between timing error signal and the residual input signal. Eventually, the loop can even fail to converge if the timing error signal is completely defiled by the residual input signal. As shown in fig.(3-7), for a gain mismatch of -20dB (10%) the calibration loop will no longer have negative feedback and will diverge. Hence, it is important to remove the gain mismatch of the two reference lanes. A simple foreground calibration of the two reference lanes would be sufficient. As these lanes are not always used, they can be easily taken offline for foreground calibration of gain mismatch. To achieve 5fs timing accuracy the gain mismatch between the two reference lanes should be better than 80dB (0.01%). Hence, foreground calibration of the two reference lanes is mandatory before they can be employed for detection of timing errors.

Similar to gain mismatch, even the input offset can be deterrent to loop convergence. For sake of

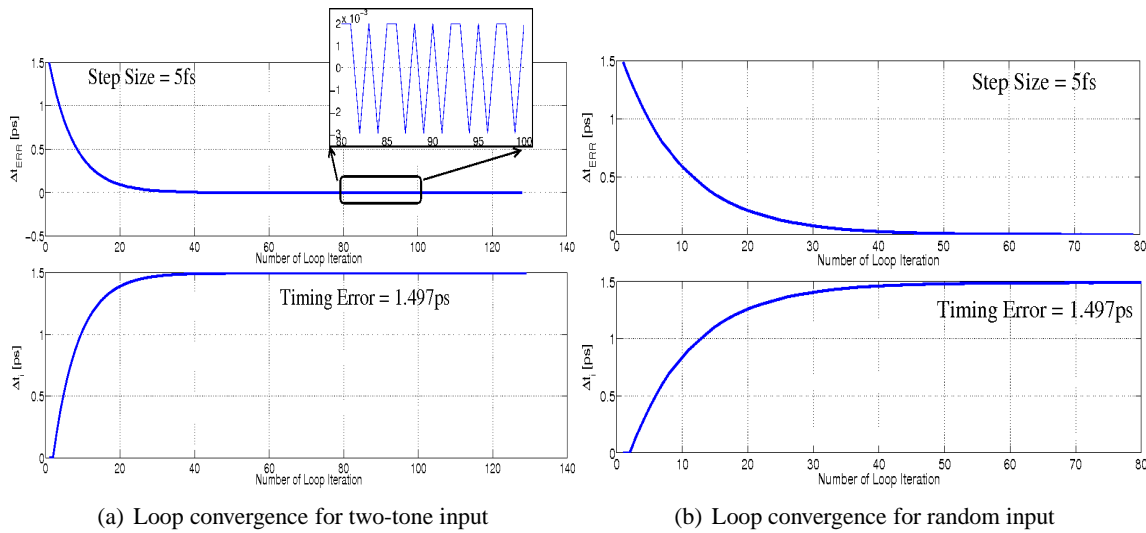


Figure 3-6: Convergence of timing error calibration loop for different input signal statistics. ($\mu_t = 1/(2e7)$)

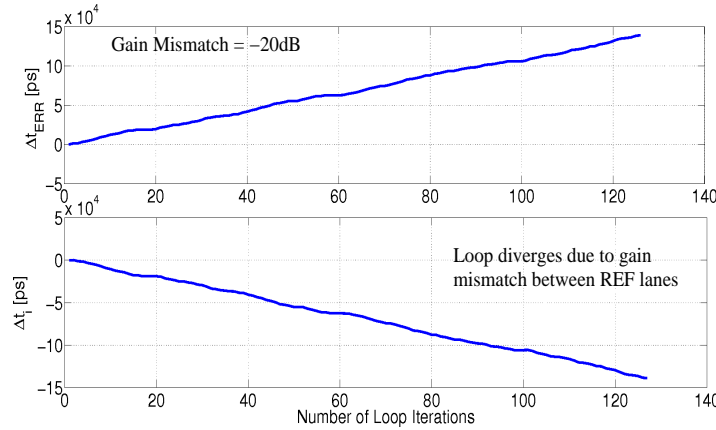
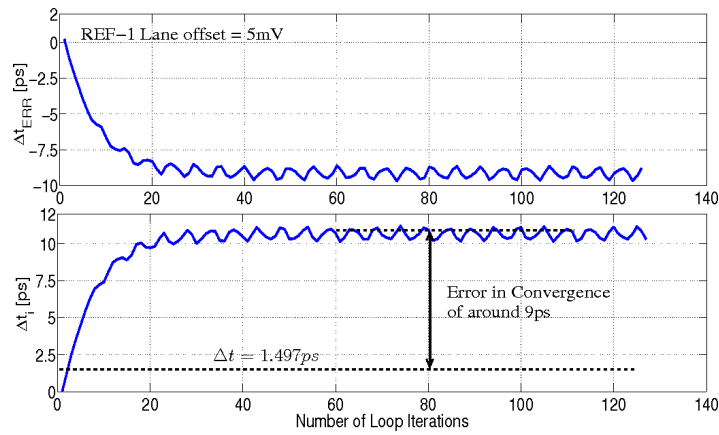
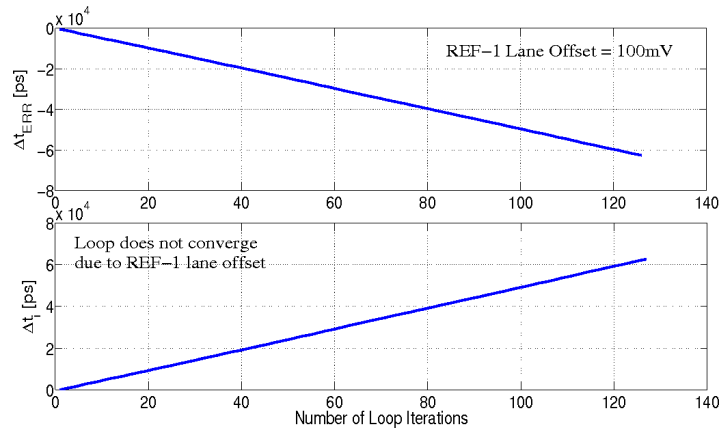


Figure 3-7: Gain mismatch between two reference lanes.

simplicity, assume that the input offset of REF-1 lane is modeled. Fig.(3-8)(a) depicts that even small offset values, in the order of 5mV are sufficient to create huge error, like 9ps, in loop convergence. Whereas, the maximum tolerable input offset to achieve timing errors of 5fs is in the order of 2nV. Further, if the offset is larger, say a 100mV, then the loop may not converge altogether. The heart of this problem is that the input offset of the reference lane appears unscathed in the signals e and D . Thus, it changes the mean of $e \cdot D$ creating a constant offset at the input of the integrator. Thus, though the negative feedback is intact, the loop cannot converge as it fails to compensate for this constant offset. Note that even the input offset of the main lane ADCs will have a similar effect. To mitigate this problem, the propagation of the offset voltage needs to be blocked before $e \cdot D$ is computed. Inserting a simple high-pass filter (HPF) in the loop, as shown in fig.(3-9)(a), will block the offset in e and D . Now the mean value of $e \cdot D$ will be only due to the timing error which the loop will drive towards zero. Fig.(3-9)(b) shows that the loop converges after inserting a high-pass filter, even if the input offset of REF-1 lane is as high as 100mV.



(a) Input offset of 5mV



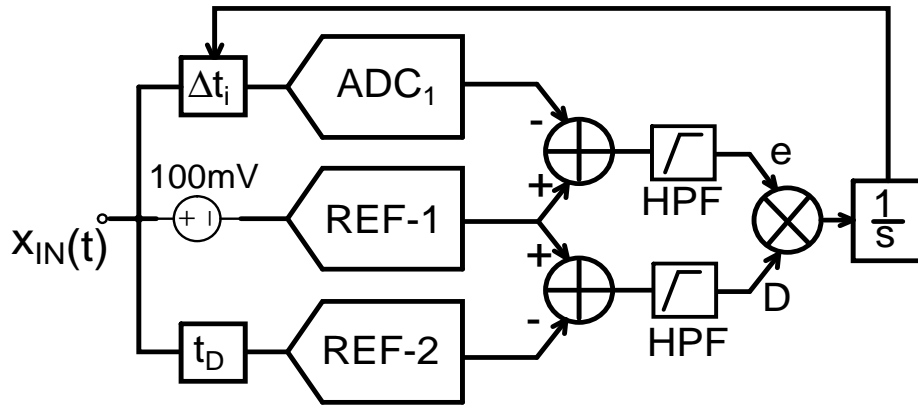
(b) Input offset of 100mV

Figure 3-8: Effect of REF-1 lane ADC's input offset on the loop convergence.

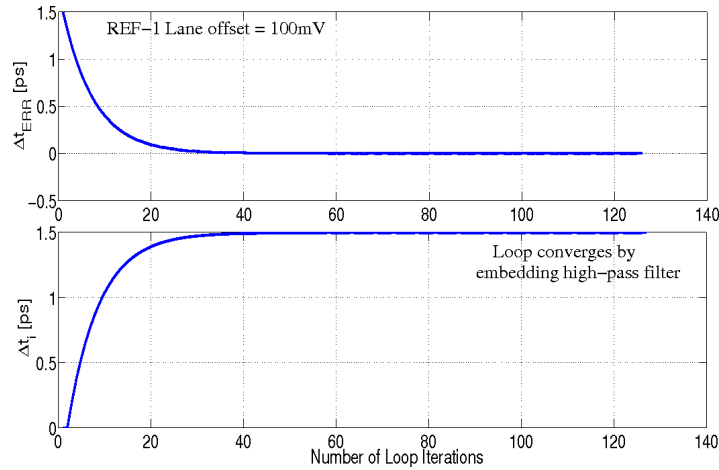
3-3-3 Limit on Speed of Convergence

The timing errors also change with temperature and supply voltage droops. The calibration loop should be fast enough to track the changes in timing error. On the other hand, the accuracy of the timing error detection can be improved by averaging the error signal across many sample. However, this slows down the loop and can potentially make it incapable of tracking changes in timing error. Thus, there is a limit on maximum number of samples available at the disposal of the calibration loop.

In order to get an estimate of this number the rate of change in temperature and supply voltage droop needs to be known. As shown in [30], the low frequency supply-voltage droop, has maximum frequency content of 100KHz. Thus, if the interleaved ADC operates at a sampling rate of 5GS/s then maximum samples within which the calibration loop should converge is around 50K. If the calibration loop takes more than 50K samples than it cannot track the changes in timing error brought by the change in supply voltage. Similarly, as shown in [31], a typical temperature sensor takes about 10KS/s to accurately track the changes in temperature. If the calibration loop is also required to track temperature at this rate then the calibration loop will have around 500K samples within which it should converge. Note that the number of samples are just an indicator.



(a) Block diagram emphasising input offset of REF-1 ADC



(b) Loop converges after embedding high-pass filter

Figure 3-9: Effect of adding high-pass filter on the convergence of the calibration loop.

3-4 Stability of Calibration Loop

The error signal, e , and the input derivative depends heavily on the amplitude and frequency of the input signal. Any change in the input signal amplitude and frequency will change $e \cdot D$ and thus the loop gain. As the stability of the calibration loop depends on loop gain, it drifts considerably with the change in amplitude and frequency of input signal. These parameters can impact the stability of the calibration loop. For instance, fig.(3-10)(a) shows the effect of varying amplitude of x_{IN} on the loop stability. If the amplitude of x_{IN} is very small it acts to over-damp the loop stability whereas if the amplitude of x_{IN} is too large it acts to under-damp the loop stability. For certain amplitude of x_{IN} (e.g. $> 3.0V$ in fig.(3-10)(b)) the calibration loop can even oscillate. In addition, value of μ_t also affects the stability of the loop. As shown in fig.(3-10)(c) with varying μ_t the response of the loop can be under-damped, critically damped or over-damped for a given input signal amplitude x_{IN} and input frequency f_{IN} . Even certain values of μ_t can result in oscillations as shown in fig.(3-10)(d). Hence, to ensure that the loop is stable across a desired range of input amplitude and frequency, the value of μ_t should be appropriately chosen. In this thesis, MATLAB simulations are used to conclude that $\mu_t = 1/(2e7)$ would ensure that the loop is stable for maximum input amplitude of $1.4V_{PP}$ and

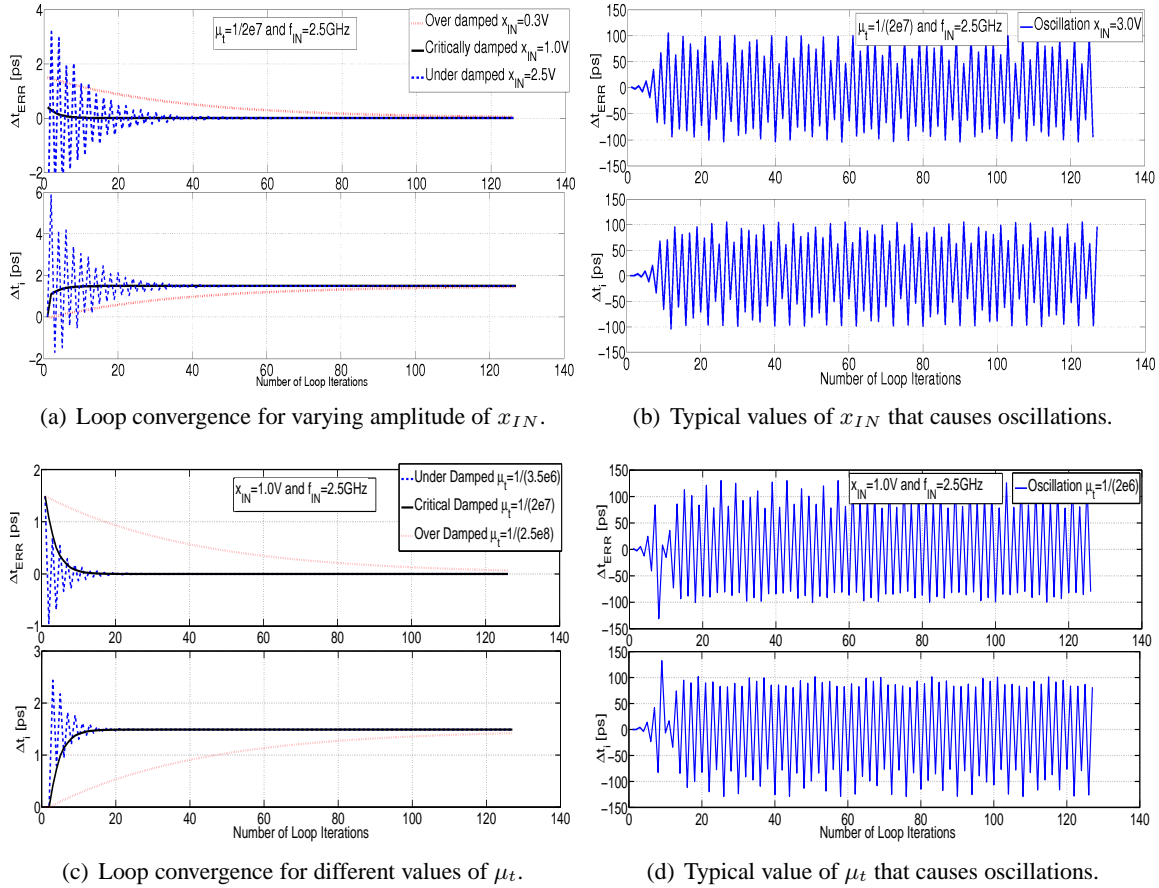


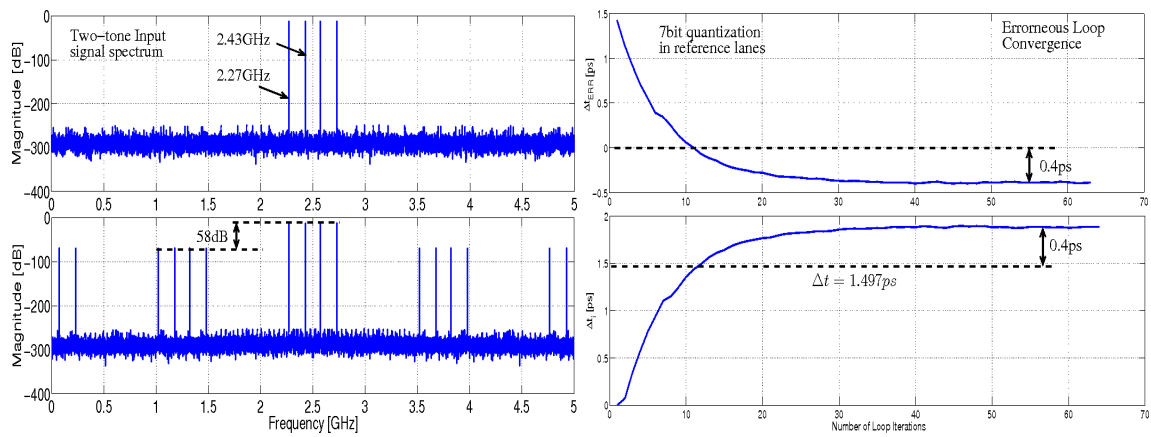
Figure 3-10: Convergence of calibration loop for different values of x_{IN} , f_{IN} and μ_t .

frequency of 2.5GHz.

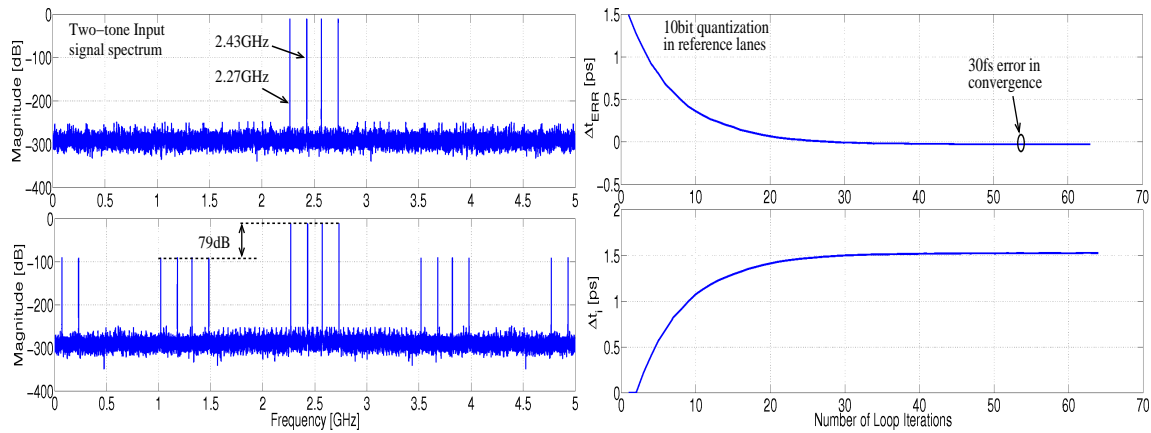
3-5 Impact of Finite Quantization of the two Reference Lanes

Lastly, the number of quantization levels in the two reference lane also hampers the convergence of the loop. Assuming a two tone input and finite quantization, the output of the two reference lanes in fig.(3-3) will have quantization error which is strongly correlated with the input. As the correlated quantization error propagates into error signal e and derivative signal D , while computing $e \cdot D$, it creates a DC offset voltage at the input of the integrator. This offset artificially shifts the mean value of $e \cdot D$ which otherwise should have been steered by the loop. This static error forces the loop to converge to a wrong steady state value. Note that this static error is independent of actual timing error, Δt and it does not change even if Δt is changed.

To illustrate this effect, assume a two-tone input signal as shown in fig.(3-11(a)) and the two reference lanes modeled with 7 bits quantization levels. Fig.(3-11(b)) shows that the timing calibration loop converges to a point which is shifted by 0.4ps from its ideal value. As a result, actual timing error ($\Delta t = 1.497ps$ in fig.(3-11(b))) is not completely canceled and spurious tones are created in the output spectrum, which are just 58dB below the two input tones (fig.(3-11(a))). Thus, finite quantization in



(a) Input and output signal spectrum when reference lanes have 7 bits quantization levels. (b) Static error when reference lanes have 7 bits quantization levels.



(c) Input and output signal spectrum when reference lanes have 10 bits quantization levels. (d) Static error when reference lanes have 10 bits quantization levels.

Figure 3-11: Effect of finite quantization of the two reference lanes on the loop convergence.

the reference lanes has a catastrophic effect on the linearity of the time-interleaved ADC.

A straightforward solution would be to reduce the amplitude of the static error (0.4ps in fig.(3-11(b)) by increasing the number of quantization levels in the reference lanes. As evident from fig.(3-11(c)-(d), this solution is indeed quite effective. Fig.(3-11(c) shows that by increasing the quantization levels from 7 bits to 10 bits the spurs due to timing error are 79dB below the two input tones. The static error is also reduced to 30fs (fig.(3-11(d)) from 0.4ps.

However, even 30fs of static error is not sufficient as the target specification on timing error (discussed in Chapter-1) is around 5fs RMS. Further increase in quantization levels to 14 bits would suffice but not without significant increase in design complexity. Alternatively, it is apparent that the static error depends upon the quantization level because the quantization error is strongly correlated to the input. The static error can be also reduced if this correlation is broken by some mechanism. In other words, the quantization error is uncorrelated to the input signal. A popular technique that exists in literature to realize this goal is to add random noise or dither along with the input signal. Fig.(3-12)(a) shows a case of adding 4 bits dither while the reference lanes have 7 bits quantization. Compared with fig.(3-11)(b) where the static error was 0.4ps, after adding dither it has now improved

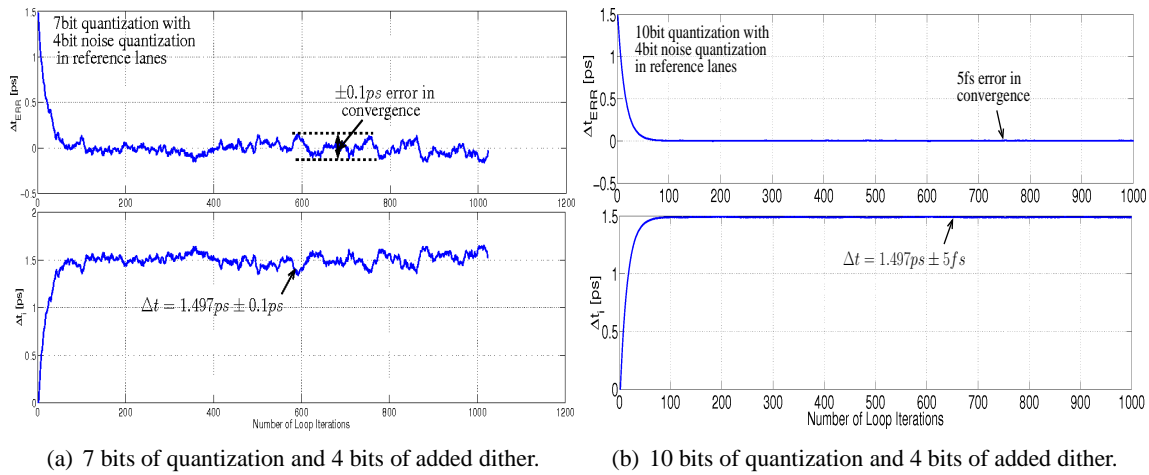


Figure 3-12: Reducing static error by adding uncorrelated dither to the input of the two reference lanes.

to $\pm 0.1ps$. Further, as evident from Fig.(3-12)(a), the final steady state value of the loop is not smooth and constant but quite noisy. To resolve this issue, before $e \cdot D$ is computed both e and D should be averaged across an increased number of samples. Though the averaging would increase the loop convergence time, it will still improve the static error as the final steady value will be less noisy. As argued in previous section, the number of available samples are limited. As a result, adding dither is not very effective as the amount of averaging that can be used is limited. A better alternative would be to use both approaches simultaneously, namely; increase the resolution of the reference lane ADCs and adding dither. As shown in Fig.(3-12)(b), the quantization level is increased to 10 bits while 4 bits of dither is simultaneously added to the input of reference lanes. Now, the static error is less than 5fs and the loop converges to actual timing error within $\pm 5fs$.

3-6 Summary

Basic operating principle of the timing error calibration loop is discussed in this chapter. Various factors that affect the convergence and stability of the loop are explored. It can be concluded that though the timing detection is independent of the input signal statistics, the convergence speed of the calibration loop still depends on it. Further, the gain mismatch between the reference lanes do affect convergence. But the reference lanes can be taken offline for foreground calibration. Similarly, input offset of reference lanes also poses threat to the loop convergence. To mitigate this problem a simple solution of inserting a high-pass filter in the loop is demonstrated with pertinent simulation results. It was also shown that the choice of LMS coefficient, μ_t , was important for stabilizing loop, particularly in the presence of varying input amplitude and frequency. A judicial choice of μ_t is critical in ensuring right balance between the stability and convergence speed of the loop. Finally, the impact of finite quantization of reference lanes on the steady state convergence of loop, is studied. A solution of adding dither is shown to suppress the static error and help the loop to achieve steady state accuracy of 5fs.

Chapter 4

Sampling-Time Error Due to Observer Effect of Reference lanes

THE input buffer isolates the actual input of an ADC from the analog circuits that drive the ADC. The sampling front-end of an ADC has switching currents and transient voltage spikes which in the absence of input buffer would be seen by the analog driving circuits. As these circuits are not designed to handle these transient switching voltages and currents, it mandates the use of an input buffer. However, the design of an input buffer is not trivial as several considerations should be taken into account. Negligible noise contribution and low distortion at Nyquist frequency are two key considerations. While a desired noise level maybe achieved by burning more power, achieving low distortion is challenging as the output impedance of the buffer needs to be low enough to drive the input impedance of an ADC. This problem is exacerbated in time-interleaved ADCs because the output of the buffer is the preferred interleaving node. As shown in fig.(4-1)(a), in conventional ADC the input buffer drives only one ADC whereas in fig.(4-1)(b) the input buffer drives time-interleaved ADCs along with reference ADCs. As no noise benefit is achieved by time-interleaved ADCs, the size of sampling capacitor for each individual ADC stays the same as that of a single ADC. Hence, due to input parasitics of the individual ADCs, the loading on the input buffer only increases for time-interleaved ADCs.

As the input buffer has finite output impedance and has increased loading due to time-interleaved ADC, the absence or presence of an additional reference lane changes its loading. This modulation of input buffer loading is noticeable on the ADCs performance (e.g. larger spurs due to timing error). This *Observer Effect* caused due to loading from reference lanes is brought out in this chapter along with its possible solutions. First, a simple estimate of the impact on timing error spurs by mere absence or presence of reference lane is derived. A simple solution to add dummy lanes is employed to mitigate this problem. Next, the transient interaction between the sampling instance of main-lane ADC and reference lane ADC is understood by the means of simulation. It will be shown that this interaction also creates timing errors. This problem can be solved by adding delay lines in the sampling front-end. Lastly, the impact of mismatch between reference lane and its dummies on timing error spurs is studied. This problem is solved by scaling the size of reference lane and dummies with respect to that of the main lane.

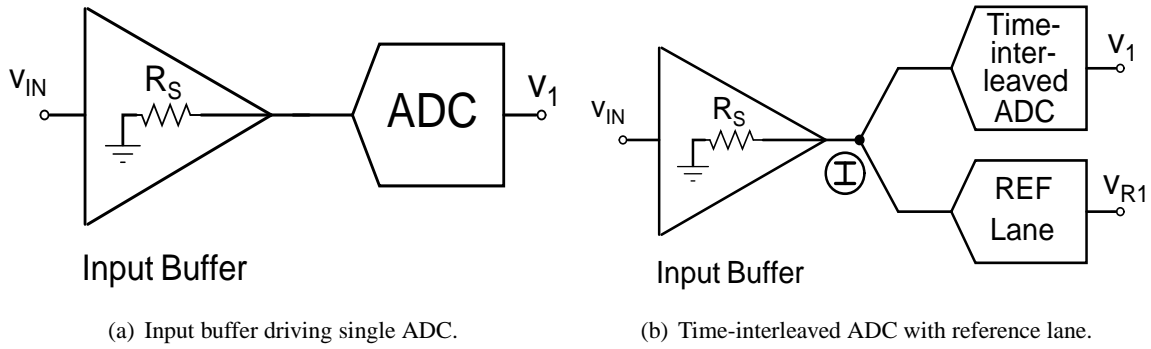


Figure 4-1: Loading of input buffer by single ADC and a time-interleaved ADC along with its reference lane.

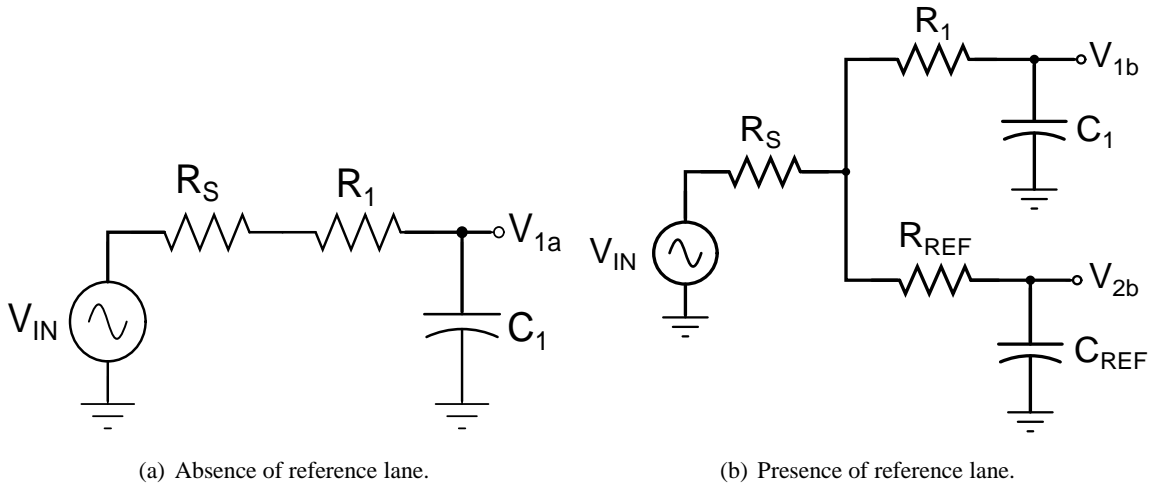


Figure 4-2: RC models for sampling front end of one out M-interleaved ADCs in the absence and presence of reference lane.

4-1 Observer Effect: Simple RC Model Analysis

In order to understand the impact of reference lane on the timing error spurs, the phase of the signal sampled by one of the M-interleaved ADCs (main-lane) should be determined both in the absence and also in the presence of a reference lane. The difference in the phase for a fixed input frequency gives the information about the timing error caused due to observer effect of the reference lane. In order to perform this exercise a simple RC model as shown in fig.(4-2)(a)-(b), is used where R_S is the output impedance of the input buffer, R_1 is the ON-resistance of the sampling switch and C_1 is the sampling capacitor of one of the M-interleaved ADCs whereas, R_{REF} and C_{REF} represent the ON-resistance of the sampling switch and sampling capacitor of reference lane, respectively. The respective outputs of the two models can be written as,

$$V_{1a} = \frac{V_{IN}}{[1 + (s(R_1 + R_S)C_1)]} \quad (4-1)$$

$$V_{1b} = \frac{V_{IN}}{[1 + (s(R_1 + R_S)C_1)] + [\frac{sR_SC_{REF}(sR_1C_1+1)}{sR_2C_{REF}+1}]} \quad (4-2)$$

In order to keep analysis simple, assume that $R_1C_1 = R_{REF}C_{REF}$. This assumption is plausible as it ensures that even if $R_1 \neq R_{REF}$ and $C_1 \neq C_{REF}$ the track-bandwidth of main-lane ADC and reference lane is equal. If the bandwidth is not equal then an additional timing error would stem from bandwidth mismatch between main-lane ADC and reference lane ADC. Under the assumption of equal bandwidth, eqn.(4-2) reduces to,

$$V_{1b} = \frac{V_{IN}}{[1 + (s(R_1 + R_S)C_1)] + [sR_SC_{REF}]} \quad (4-3)$$

Using eqn.(4-1) and eqn.(4-3), the phase of the signal sampled across C_1 in two cases can be written as,

$$\phi_{V1a} = -\tan^{-1}[\omega(R_S + R_1)C_1] \quad (4-4)$$

$$\phi_{V1b} = -\tan^{-1}[\omega(R_S + R_1)C_1 + \omega R_SC_{REF}] \quad (4-5)$$

The difference in phase is the error caused due to the observer effect of the reference lane. This difference can be written as,

$$\Delta\phi = \phi_{V1a} - \phi_{V1b} = \tan^{-1}\left[\frac{\omega R_SC_{REF}}{1 + \underbrace{\omega^2[(R_1 + R_S)C_1 + R_SC_{REF}][(R_1 + R_S)C_1]}_{\text{Ignore as } \ll 1}}\right] \quad (4-6)$$

The timing error due to observer effect of reference lane (Δt_{SI}) can be obtained by simplifying the equation above and dividing it with the fixed input frequency as shown below,

$$\Delta\phi \approx \omega R_SC_{REF} \Rightarrow \boxed{\Delta t_{SI} \approx R_S \cdot C_{REF}} \quad (4-7)$$

The equation above is a simple approximation of the actual timing error caused by the observer effect of the reference lane. In most cases, this approximation will prove quite reasonable and acceptable. Note that if $R_1C_1 \neq R_{REF}C_{REF}$ then the eqn.(4-7) will have additional terms. However, the dominant term will be still R_SC_{REF} . The eqn.(4-7) can also be derived by applying Elmore's delay formula between V_{IN} and V_{1b} in fig.(4-2)(b). (To learn more about Elmore's delay, refer to [32])

To get an estimate of the timing error Δt_{SI} , assume $R_S = 10\Omega$ and for 12 bit thermal noise level at $80^\circ C$ temperature, the sampling capacitor C_1 should be at least $1.4pF$ resulting in Δt_{SI} of $14ps$. Thus, mere absence and presence of reference lane can create large timing errors.

Further, this scenario is indeed realistic because the time-interleaved ADC typically works in two phases, viz; calibration phase and normal operation phase. In calibration phase the reference lane is used to detect timing errors. Once timing errors are corrected, the reference lane is pulled out and the time-interleaved ADC operates in normal operation phase. To avoid creating timing errors due to observer effect of reference lane, one elementary solution is to insert a dummy sampling circuit which resembles the sampling front-end of the reference lane. Fig.(4-3) demonstrates this concept where

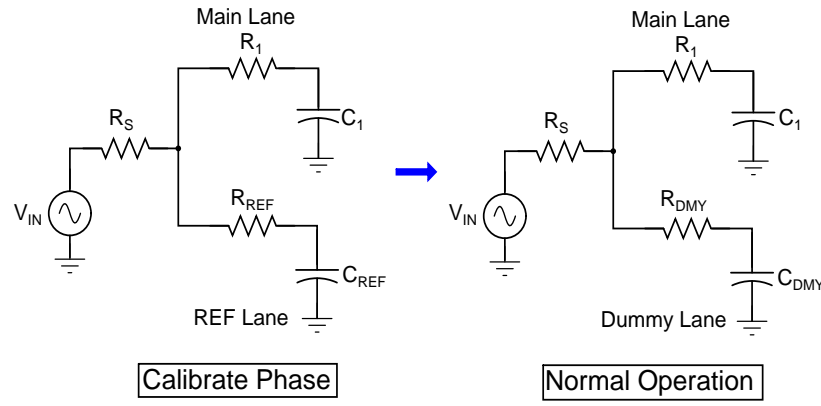


Figure 4-3: Two phases of operation for time-interleaved ADC: a) Calibration phase and b) Normal phase of operation.

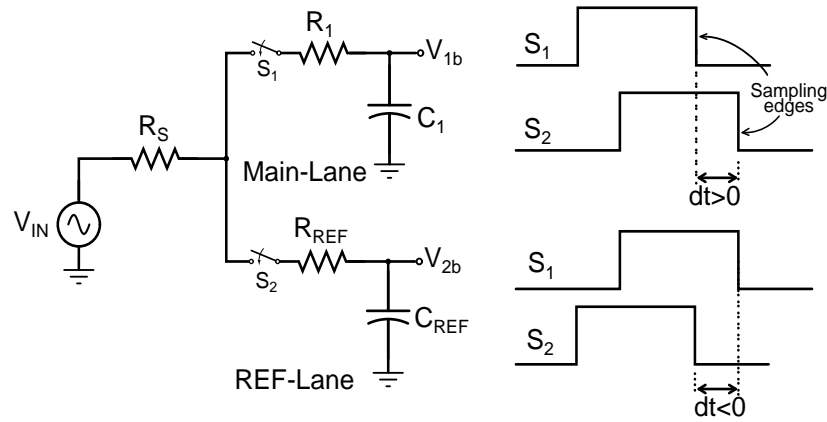


Figure 4-4: Estimation of input signal derivative by using additional reference lane with delayed input.

reference lane loads the input buffer during the calibration phase but during the normal operation phase its loading is maintained by a dummy sampling front-end. Otherwise in the absence of reference lane loading a timing error of Δt_{SI} is created. Thus the observer effect of reference lane can be mitigated by inserting a dummy sampling front-end.

From eqn.(4-7) it can be inferred that the observer effect of reference lane can be avoided by making $R_S = 0$. Though it is precisely true, making $R_S = 0$ at Nyquist frequencies would require an input buffer with very high bandwidth incurring heavy power consumption penalty. Even achieving $R_S = 10\Omega$ at input frequencies of 2.5GHz is a non-trivial task.

4-2 Observer Effect: Sampling Instance Interactions

The observer effect of reference lanes is not limited only to the modulation of input buffer loading. The time-shift between the sampling instances of the main-lane and the reference lane can also create timing errors. To investigate this timing errors consider a circuit shown in fig.(4-4) where, R_1 and C_1

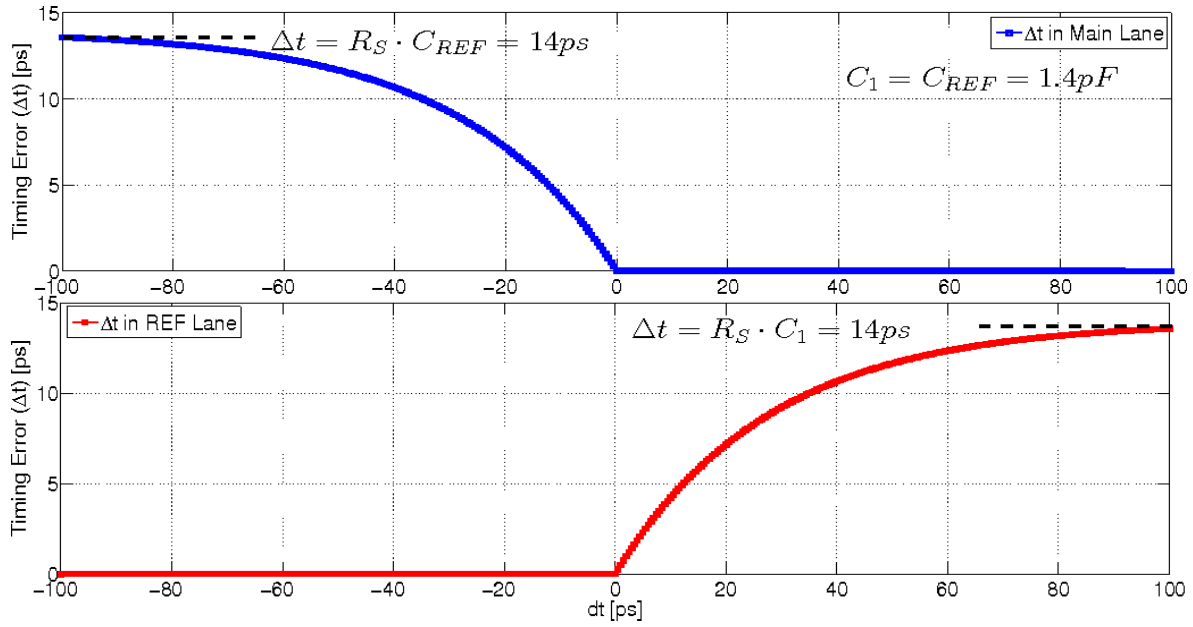


Figure 4-5: Simulation of interaction between main-lane and reference lane sampling.

model the sampling front-end of the main-lane ADC whereas, R_{REF} and C_{REF} model the sampling front-end of the reference lane ADC. S_1 and S_2 are their respective sampling edges. As shown in fig.(4-4), the time shift between the two sampling edges is dt and, for the sake of argument, it is with respect to the main-lane sampling edge S_1 . Thus, for $dt < 0$, the reference lane samples the input signal before the main-lane and for $dt > 0$, the main-lane samples the input before the reference lane. Now, if $dt < 0$ then the loading of the input buffer would change before the main-lane can sample the input. Though, the main-lane would notice this change in loading delayed through $R_1 C_1$, it would be sufficient to cause timing error. Similar explanation holds for $dt > 0$ where the main-lane samples before the reference lane. Now, even though the sampling time error is created in the reference lane, it will affect the main-lane ADC through the feedback loop. On the other hand, no timing error is created for $dt = 0$. Note that in any case the timing error caused by this interaction will not be as large as Δt_{SI} (eqn.(4-7)).

Fig.(4-5) shows the simulation result for timing error, Δt , created by both $dt < 0$ and $dt > 0$. The required simulation setup is provided in Appendix-B. For $dt < -100ps$ i.e. reference lane samples $100ps$ earlier than the main-lane, the timing error is equal to Δt_{SI} . This is because the tracking time constant of main-lane sampling circuit is about $28ps$ ($R_S = 10\Omega$, $R_1 = 10\Omega$, $C_1 = 1.4pF$) which is much less than $100ps$. Hence, the main-lane perceives no reference lane while it samples the input. This is equivalent to the load modulation effect expressed by eqn.(4-7). Similarly, if $dt > 100ps$ then the reference lane will not see any loading due to main-lane and again the timing error would be equal to Δt_{SI} . However, as the dt approaches zero from both the sides i.e. from $-100ps$ and also from $+100ps$, the timing error drops exponential for main-lane and reference-lane sampling, respectively.

As evident from eqn.(4-7), the timing error depends strongly on the value of sampling capacitor C_{REF} for the main-lane (C_1 for the reference lane). An immediate idea would be to scale the size of the sampling capacitor. Assume that the size of C_{REF} is scaled then its effect on the timing error in the main-lane and in the reference lane is as shown in fig.(4-6). The timing error in main-lane reduces

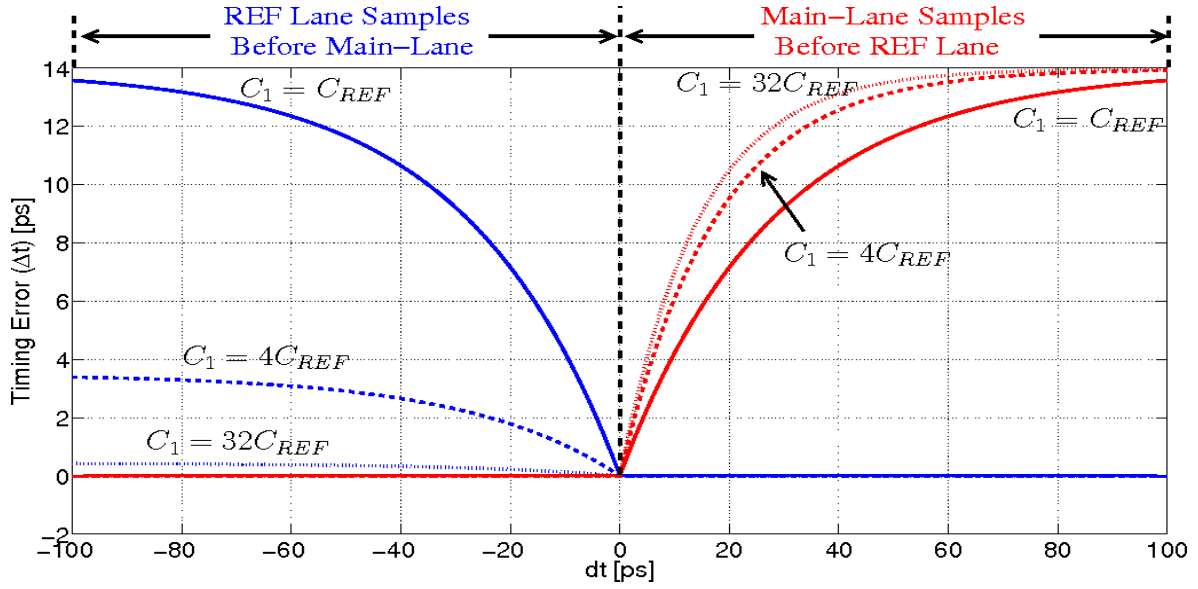


Figure 4-6: Effect of C_{REF} on the interaction between main-lane and reference lane sampling.

as the size of C_{REF} is reduced but it increases for the input signal sampled in the reference lane.

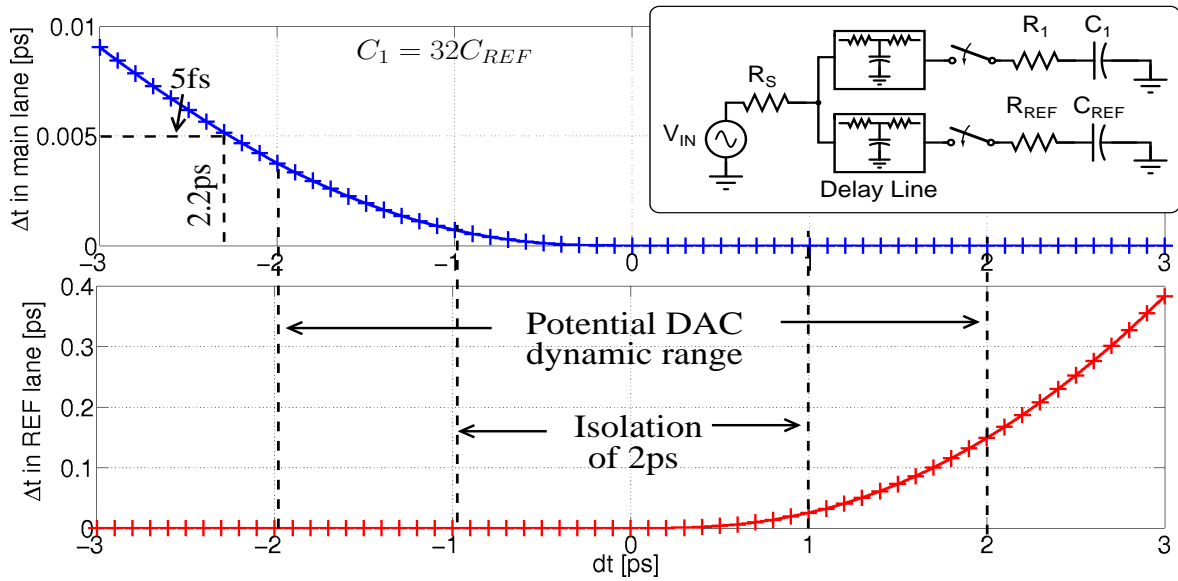
4-3 Isolating the Sampling Interactions using a Wire Delay

The timing error calibration loop adjusts the clock-edges to compensate for the timing errors. The interaction between sampling time instances is an additional source of timing error due to loading (observer) effect of the reference lane. Due to this interaction the calibration loop detects wrong value of the timing error. Hence, it also corrects for a wrong value of the timing error. As a result, the final converged steady-state value of the calibration loop is erroneous which is unacceptable. Thus, it would be highly desirable to suppress the interaction between the sampling instances of the main-lane and the reference lane.

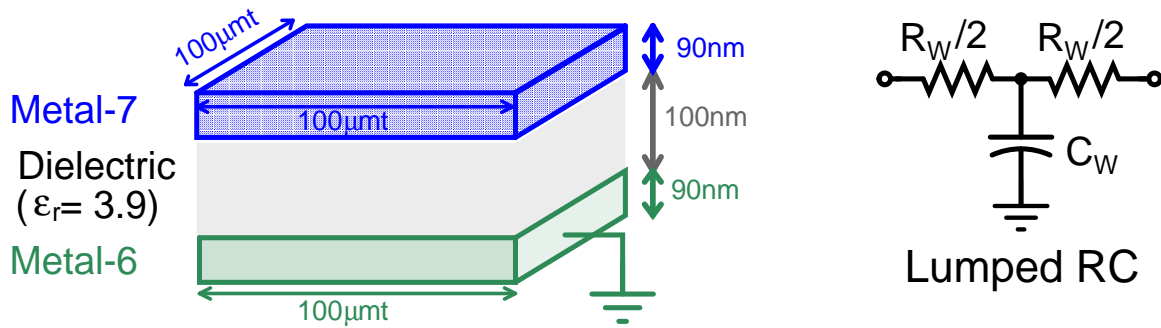
A plausible solution would be to employ some method to isolate this interaction between the sampling instances. The isolation should be such that the sampling time interaction is at least not present in the operation range of the calibration loop. One way to achieve this goal is to insert delay lines in the sampling front-end of main-lane and the reference lane. The inset picture in fig.(4-7)(a) depicts this solution. Assuming, the delay line adds 1ps delay, the resulting timing error would be as shown in fig.(4-7)(a). The two interactions are now sufficiently isolated creating a space to accommodate the dynamic range of the DAC. Now the question is how to design this delay line. As it will be in the input signal path, it should meet strict noise and distortion specifications. The simplest approach to this problem would be to use wire delay to implement a delay line.

Assuming that the delay line is modeled by a lumped RC model as shown in fig.(4-7)(b), the delay of a wire is given by,

$$t_d = \frac{R_W \cdot C_W}{2} \quad \text{where, } R_W = \left(\frac{L}{W}\right) \cdot R_{\square} \quad \text{and} \quad C_W = c_w \cdot WL \quad (4-8)$$



(a) Isolating the sampling-time interaction by using delay lines.



(b) Estimating wire delay in industrial 28nm CMOS process.

Figure 4-7: Using wire based delay lines to isolate interaction between main-lane and reference lane sampling instances.

The equation above can be simplified into a relation between wire-length and wire-delay,

$$L = \sqrt{\frac{2 \cdot t_d}{R_{\square} \cdot c_w}} \quad (4-9)$$

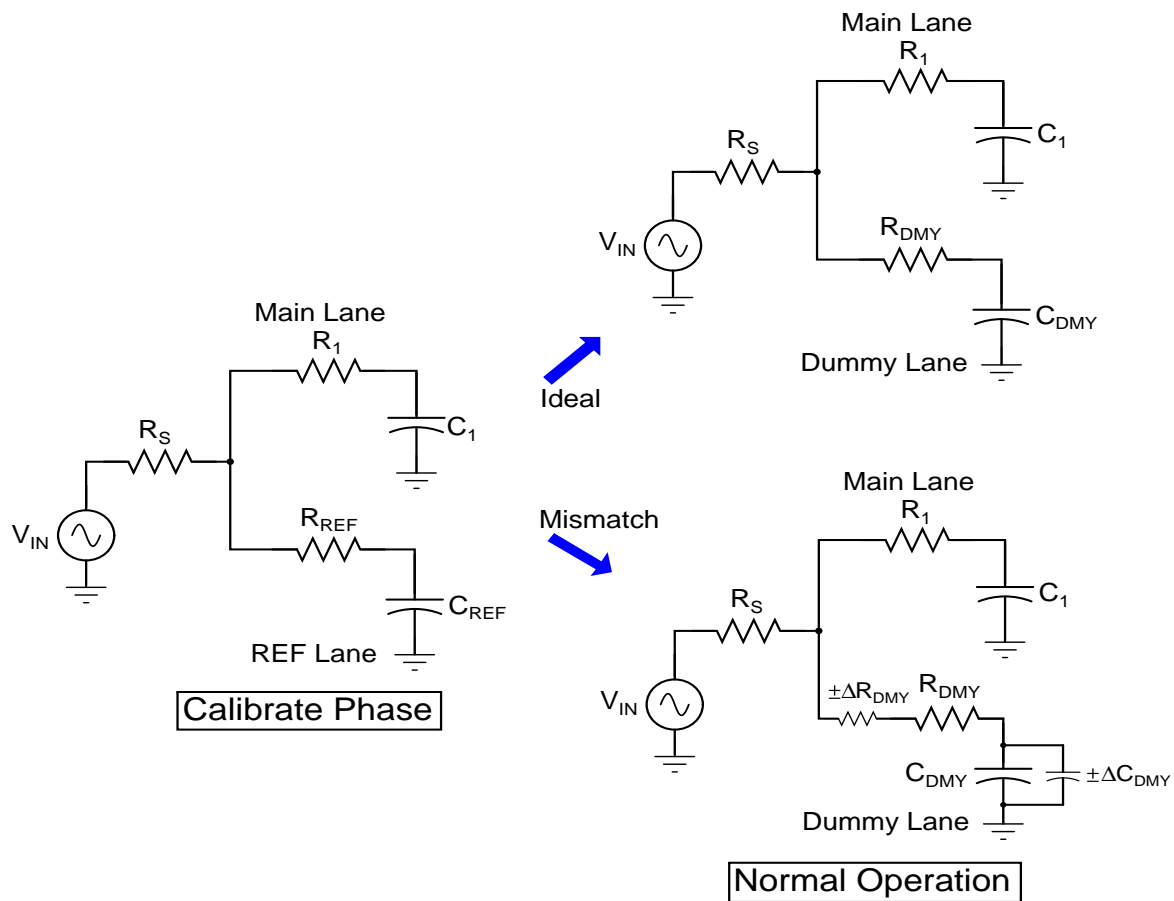
The value of sheet resistance, R_{\square} for metal-7 (Copper) in industrial 28nm CMOS process is about $0.53\Omega/\square$. The unit capacitance c_w is estimated by extracting it from the structure shown in fig.(4-7)(b). Its value is $0.263fF/\mu m^2$ and it is accurate within $\pm 5\%$. Using these values and assuming $W = 10\mu m$ in eqn.(4-9), to achieve wire-delay, t_d of 1ps the length of the wire should be about $119\mu m$. Note that the width of the wire does not play a role in eqn.(4-8) as it factors out in the product of R_W and C_W . Nevertheless, for small width fringe capacitance is dominant and the model assumed above will no longer hold valid [33]. Lastly, a quick sanity check on the value of length calculated using eqn.(4-9) can be performed by estimating distance traveled by at speed of light ($c = 3 \times 10^8 m/s$). The relative permittivity for silicon-dioxide is 3.9. Thus, light would travel with speed of $1.5 \times 10^8 m/s$. In time delay of 1ps, a wave of light would cover a distance of $150\mu m$ whereas, the value predicted by eqn.(4-9) is $119\mu m$. The discrepancy arises because the eqn.(4-9) is a simple lumped approximation. If the electrons were to travel at speed of light then many other physical effects (e.g. skin depth) should be considered to accurately compute the delay. This will only lead to more complicated equation which would not be useful for quick hand-calculations. Thus, looking at the simulation results and by quick sanity check, the eqn.(4-9) seems correct.

4-4 Mismatch between Dummy lane and REF lane

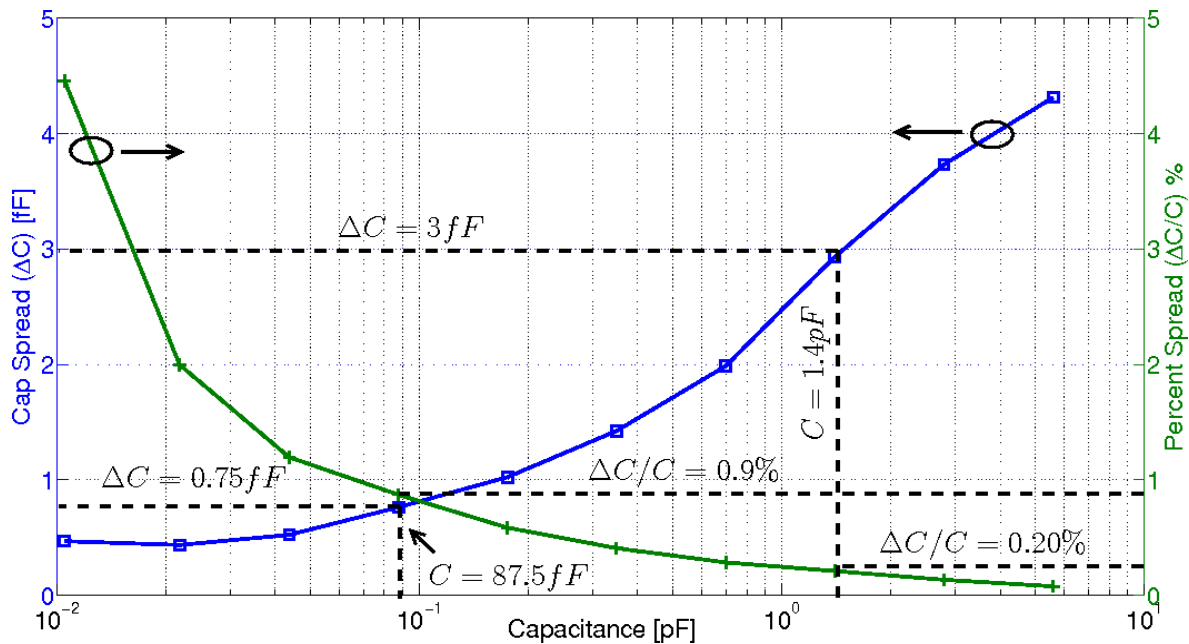
As argued previously, the time-interleaved ADC works in two phases, namely; calibration phase and normal operation phase. In order to remove timing errors due to observer effect of reference lane, fig.(4-3) gave a solution of replacing reference lane with dummy lane during normal operation phase. However, the dummy lane is not exactly equal to the sampling front-end of the reference lane as it merely tries to resembles it. In this section, the impact of mismatch between dummy lane and the reference lane on timing mismatch is studied, and solutions to suppress its impact are investigated.

Fig.(4-8)(a) highlights the problem of mismatch between reference lane and dummy lane. Under ideal condition, the dummy lane matches exactly to the reference lane. Thereby, the value of ON-resistance of sampling switch and the sampling capacitor for both reference lane and dummy lane are exactly equal, i.e. $R_{REF} = R_{DMY}$ and $C_{REF} = C_{DMY}$. But, in the presence of mismatch the dummy lane has slightly different values for ON-resistance of the sampling switch and the sampling capacitor, i.e. $R_{REF} = R_{DMY} \pm \Delta R_{DMY}$ and $C_{REF} = C_{DMY} \pm \Delta C_{DMY}$. Unfortunately, all CMOS process are inherently mismatch prone and the present industrial 28nm CMOS process is no exception. Hence, the dummy lane will be slightly different from the reference lane. The critical questions however remains is 1) how does the mismatch impact timing error and 2) how to minimize it?

As evident from the eqn.(4-7), the loading of reference lane primarily affects timing error through its sampling capacitor. The ON-resistance of the sampling switch does not play a crucial role at least to a first order. The same argument can be extended for dummy lane as its mismatch manifests into timing error through the source impedance R_S . Thus, $\pm \Delta R_{DMY}$ is not critical and can be ignored. The important factor is the mismatch in the sampling capacitor, $\pm \Delta C_{DMY}$. Further, the timing error would be larger for larger values of ΔC_{DMY} . As apparent from fig.(4-8)(b) a capacitor of 1.4pF



(a) Mismatch effect of dummy lane



(b) Absolute spread and percentage spread in capacitor value for industrial 28nm CMOS process.

Figure 4-8: Impact of mismatch between dummy lane and reference lane, and its dependence on the size of the sampling capacitor.

spreads by 3fF whereas a capacitor which is 16x smaller, 87.5fF spreads only by 0.75fF. Whereas, the percentage spread is much better for 1.4pF than for 87.5fF. Hence, to minimize ΔC_{DMY} a smaller sampling capacitor should be used but as the accuracy of a bigger capacitor is better it should be used for main-lane ADC. Thus, the sampling capacitor of the dummy lane affects the timing error and it should be made smaller to minimize its impact on timing error.

The argument above is intuitive but not quantitative. It would be useful to have an equation which can estimate the timing error due to a given spread in the sampling capacitor of dummy lane. To achieve this goal, first assume a single ended circuit as shown in fig.(4-9)(a). The spread in C_1 (σ_{C1}) would cause spread in track-bandwidth and hence the timing error. The impact of σ_{C1} on the timing error can be calculated from the resulting spread in phase. Assume that the spread in C_1 is Normally distributed and it is expressed as $\mathcal{N}(\mu_{C1}, \sigma_{C1})$. Thus, the spread in the phase of a single-end sampling circuit is also Normal and it can be expressed as,

$$\hat{\phi}_{SE} \sim \mathcal{N}(\underbrace{\mu_{SE}}_{\text{Mean}}, \underbrace{\sigma_{SE}}_{\text{Variance}}) \quad (4-10)$$

As the focus is on computing the timing error, only the variance of phase, σ_{SE} , needs to be computed. It can be written as,

$$\sigma_{\hat{\phi}_{SE}} = \tan^{-1}(\omega_1(R_S + R_1) \cdot \sigma_{C1}) \quad (4-11)$$

$$\approx \omega_1(R_S + R_1) \cdot \sigma_{C1} \quad (\because \tan^{-1}(\theta) \approx \theta) \quad (4-12)$$

where, ω_1 is the fixed input frequency. The timing error is,

$$\Delta t_{SE} = \frac{\sigma_{\hat{\phi}_{SE}}}{\omega_1} = (R_S + R_1) \cdot \sigma_{C1} \quad (4-13)$$

For $R_S = 10\Omega$, $R_1 = 10\Omega$ and $\sigma_{C1} = 3\text{fF}$, the timing error, Δt_{SE} is about 60fs. This value can be verified from simulation result in fig.(4-9)(b) where the spread in Δt_{SE} is shown due to spread in C_1 over 50 Monte-Carlo runs. The simulations shows Δt_{SE} of 60.3fs. Further, as the actual track-and-hold used in the design is a fully-differential circuit, this simple analysis should be extended to a differential circuit (fig.(4-9)(a)) also. To derive this expression, assume two differential input signals as,

$$V_1 = V_{IN} \cdot \sin(\omega_1 t + \hat{\phi}_1) \quad (4-14)$$

$$V_2 = V_{IN} \cdot \sin(\omega_1 t + 180^\circ + \hat{\phi}_2) \quad (4-15)$$

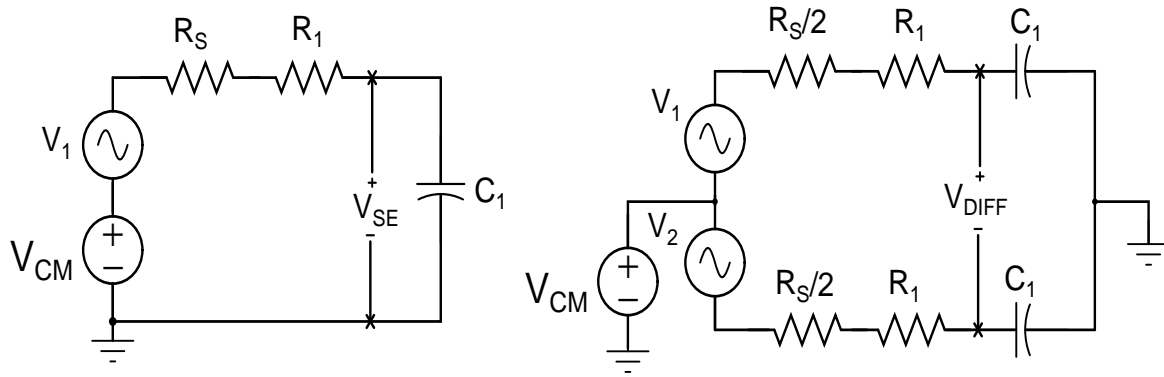
Similar to $\hat{\phi}_{SE}$, $\hat{\phi}_1$ and $\hat{\phi}_2$ represent spread in phase due to spread in C_1 .

$$\hat{\phi}_1 = \hat{\phi}_2 = \hat{\phi}_{SE}$$

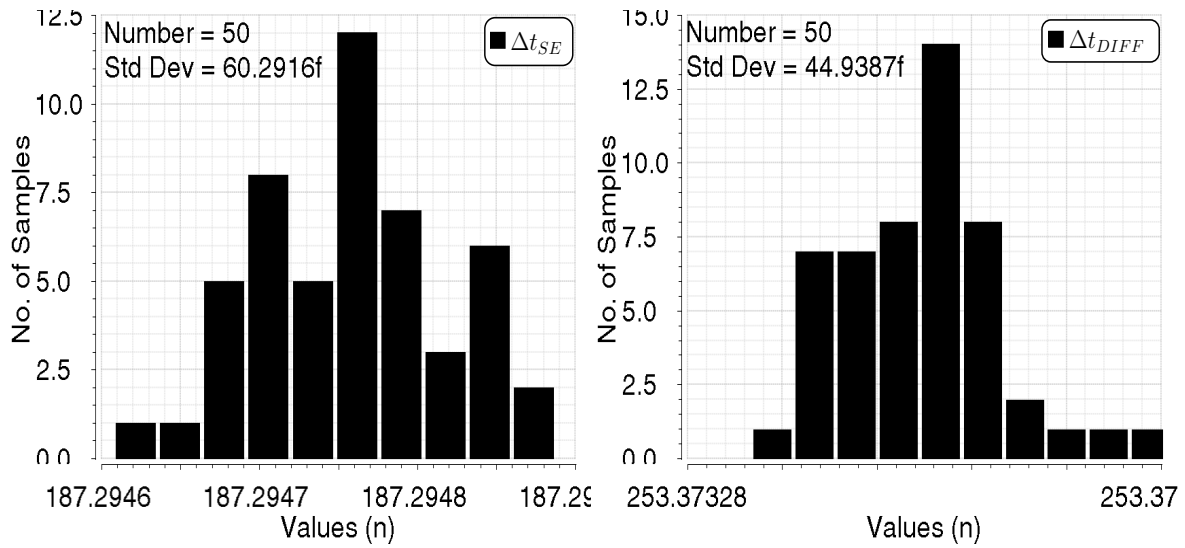
Now, the differential output voltage, V_{DIFF} can be written as,

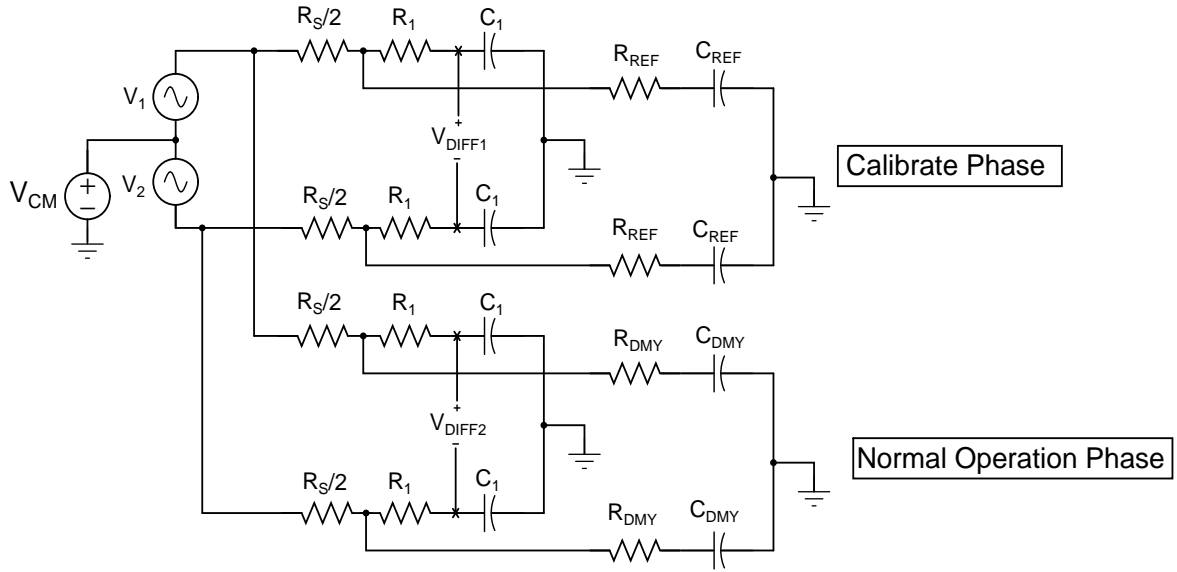
$$V_{DIFF} = V_1 - V_2 \quad (4-16)$$

$$= 2V_{IN} \cdot \sin(\omega_1 t + \frac{\hat{\phi}_1 + \hat{\phi}_2}{2}) \cos(\frac{\hat{\phi}_1 - \hat{\phi}_2}{2}) \quad (4-17)$$

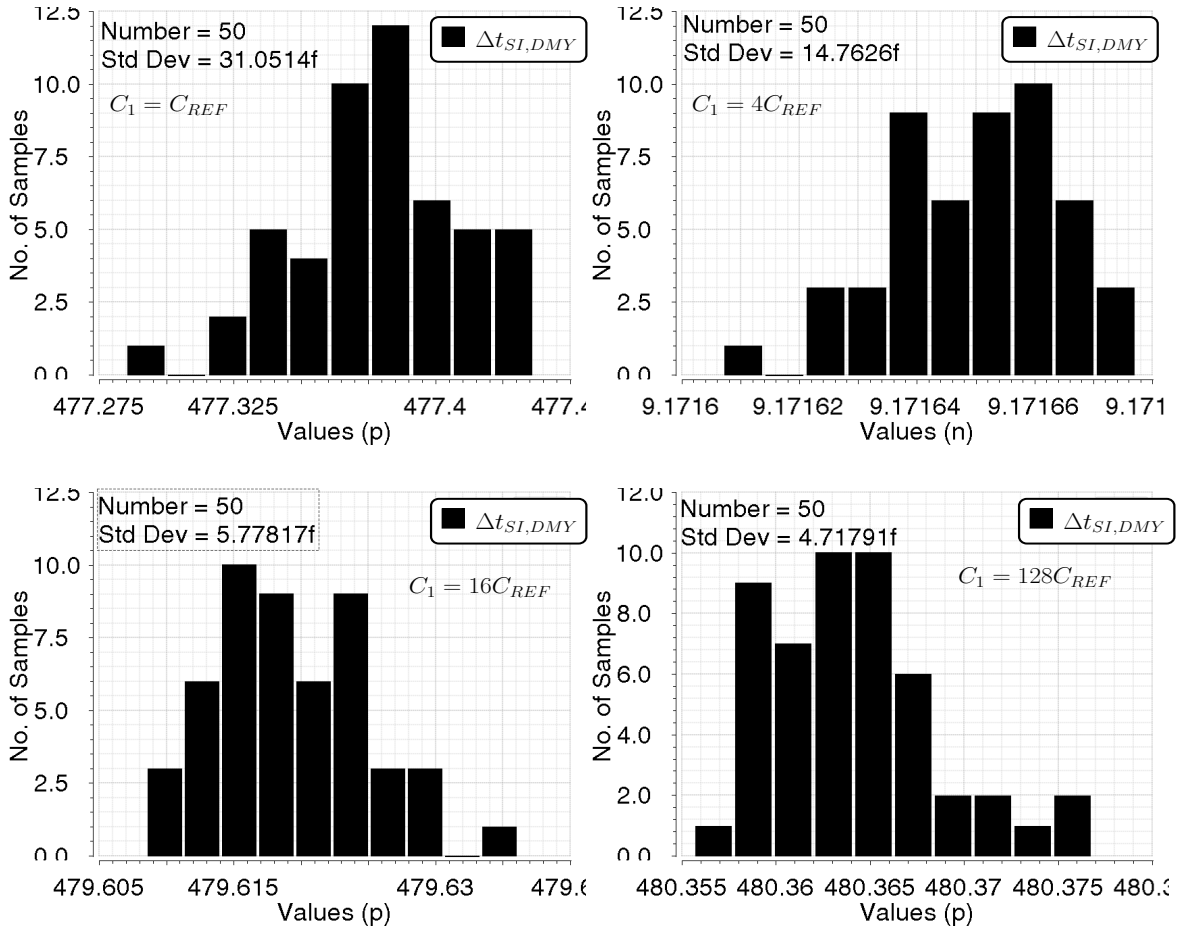


(a) RC model of single-ended and differential track-and-hold circuit.

(b) Impact of spread in C_1 on timing mismatch for single ended and differential track-and-hold circuit.**Figure 4-9:** Validating eqn.(4-13) and eqn.(4-20) for spread in C_1 .



(a) Circuit used to simulate mismatch between reference lane and dummy lane

(b) Spread in $\Delta t_{SI,DMY}$ for different values of C_{REF} .**Figure 4-10:** Minimizing timing error due to mismatch between reference lane and dummy lane by scaling the sampling capacitor.

The $\cos(\frac{\hat{\phi}_1 - \hat{\phi}_2}{2})$ term modulates the amplitude. However, as the spread in phase is small (in the order of $\mu\text{degrees}$) it can be approximated to 1 ($\because \cos(0) = 1$). Similar to the single ended case, the variance in the phase can thus, be estimated as,

$$\sigma(\frac{\hat{\phi}_1 + \hat{\phi}_2}{2}) = \frac{\sqrt{2} \cdot \sigma_{\phi\hat{S}E}}{2} = \frac{\sigma_{\phi\hat{S}E}}{\sqrt{2}} \quad (4-18)$$

The timing error can be extracted from the equation above and it can be written as,

$$\Delta t_{DIFF} = \frac{\sigma((\hat{\phi}_1 + \hat{\phi}_2)/2)}{\omega_1} \quad (4-19)$$

Using eqn.(4-13) and eqn.(4-18) with equation above, the timing error in fully differential sampling front-end is written as,

$$\Delta t_{DIFF} = \frac{\Delta t_{SE}}{\sqrt{2}} = \frac{(R_S + R_1) \cdot \sigma_{C1}}{\sqrt{2}} \quad (4-20)$$

Note that the timing error in a fully-differential circuit is $1/\sqrt{2}$ times smaller compared with single ended circuit. For $R_S = 10\Omega$, $R_1 = 10\Omega$ and $\sigma_{C1} = 3\text{fF}$, the timing error, Δt_{DIFF} is about 42.4fs. The simulation result in fig.(4-9)(b) show that Δt_{DIFF} is 44.9fs which is close to the value predicted by eqn.(4-20).

As a last step, the equation for timing error due to mismatch between reference lane and dummy lane needs to be derived. Eqn.(4-7) states that the effect of reference lane loading on the timing error is only through R_S . This argument can be applied to eqn.(4-20). The effect of mismatch between reference lane and dummy lane will affect timing error in the main-lane through R_S . This argument is only true to the first order and is not precise. Using eqn.(4-7) and eqn.(4-20) the equation for timing error due to mismatch in dummy lane can be written as,

$$\Delta t_{SI,DMY} = \frac{R_S \cdot \sigma_{CDMY}}{\sqrt{2}} \quad (4-21)$$

where, σ_{CDMY} is the spread in the sampling capacitor in the dummy lane. From fig.(4-8)(b), for sampling capacitor $C_1 = 1.4\text{pF}$ the spread is $\sigma_{C1} \approx 3\text{fF}$. Using these values in the eqn.(4-21), results in a timing error of 22fs. The sanity of this equation can be verified by using a simulation setup shown in fig.(4-10)(a). Two sampling front-end of main-lane ADC are shown with one loaded by reference lane and the other by the dummy lane. The timing error due to mismatch between reference and dummy lane can then be obtained by looking at the phase of V_{DIFF1} and V_{DIFF2} . Fig.(4-10)(b) shows the simulation result of $\Delta t_{SI,DMY}$ over 50 Monte-Carlo runs. It can be seen that $\Delta t_{SI,DMY}$ for $C_S = C_{REF}$ is 31fs. Though this value slightly differs from hand-calculated (22fs), it is still reasonable considering the fact that eqn.(4-21) is merely an approximation. To establish the sanity of eqn.(4-21) few quick hand-calculations for other simulation results of fig.(4-10)(b) can be performed. Say, the reference lane capacitor is scaled by 16x to $C_{REF} = 87.5\text{fF}$ with spread of $\sigma_{CREF} = \sigma_{CS}/4 = 0.8\text{fF}$. Using these values in eqn.(4-21), results in timing error $\Delta t_{SI,DMY}$ of 5.6fs. Alternatively, from fig.(4-8)(b) for $C_{REF} = 87.5\text{fF}$, its spread is $\sigma_{CREF} = 0.75\text{fF}$ which results in $\Delta t_{SI,DMY}$ of 5.3fs. Both of these hand-calculated values can be verified from fig.(4-10). The value for $\Delta t_{SI,DMY}$ is 5.77fs for $C_1 = 16C_{REF}$. Similarly, the eqn.(4-21) also holds for remaining two simulation results.

Finally, note that the variance in $\Delta t_{SI,DMY}$ does not exactly scale by a factor of $\sqrt{8}$ while the value of C_{REF} is scaled from $16C_{REF}$ to $128C_{REF}$. This phenomenon is due to imperfect modeling of capacitance particularly at very small values. For example consider a capacitance of 10fF in fig.(4-8)(b), the corresponding spread is about 4.5%. However, when the capacitance is increased by 4x to 40fF the spread in capacitance goes down to 1.5%. This not exact scaling of 2x. Thus, as the small capacitance are not model perfectly they exhibit considerable amount of spread which is highlighted in fig.(4-10)(b).

In conclusion, from the simulation results of fig.(4-10)(b), the reference lane should be at least scaled by a factor of 16x to keep the timing error due to mismatch between reference lane and dummy lane down to 5fs level.

4-5 Summary

Summarizing, in this chapter the observer effect of the reference lanes was studied and potential solutions are discussed to mitigate its effect. The reference lanes change the loading of the input buffer which gives rise to the observer effect. This effect creates additional timing errors and it manifests itself in two ways. First, mere absence or presence of reference lane creates huge timing errors depending upon the value of the source impedance (input buffer) and sampling capacitor. Second, the sampling instance of the reference lanes and main-lane are not exactly aligned also creates timing error. To solve the first observer effect dummy lanes are introduced during the normal operation phase, whereas the second form of observer effect is solved by inserting delay lines in the sampling front-ends. The delay lines can be implemented using simple metal wires. A simple model to compute the length of these metal wires is also provided. Finally, the mismatch between the dummy lanes and the reference lanes was identified as a source of a timing error. It was shown that to reduce this timing error below 5fs level, the reference lanes and the dummy lanes should be scaled 16x smaller compared with the main lane.

Chapter 5

Circuit Implementation

THIS chapter describes the implementation details of the timing error calibration loop for 2x-interleaved ADC. The design strategies and circuit techniques introduced in previous chapters are employed and their effectiveness is demonstrated through simulations. All the circuits are designed using industrial 28nm CMOS process and simulated in Cadence Spectre tool. The chapter starts with a system-level overview of the calibration loop. It further proceeds to discuss the implementation details of the sub-blocks in the calibration loop, including auxiliary circuits like the replica bias circuits and non-overlapping clock generation. The non-ideal behavior of some circuit blocks may hamper the loop convergence. At such instances, MATLAB simulations are also resorted. The chapter ends with a summary of important specifications achieved by each sub-blocks.

5-1 System-Level Design

A system-level diagram of the timing error calibration loop for 2x time-interleaved ADC is illustrated in fig.(5-1). It consists of two main-lane ADCs, ADC_1 and ADC_2 , interleaved together. The two reference lane ADCs used for detecting the timing errors are $REF - 1$ and $REF - 2$. The delay element, t_D delays the clock of the $REF - 2$ ADC in order to estimate the derivative of the input signal. The off-chip block (highlighted in green) was implemented using MATLAB. Rest of the other blocks (highlighted in blue) are the sub-blocks whose circuit design details are covered in the succeeding sections. In the subsequent sections circuit design details of sub-blocks like: track-and-hold for main and reference lanes, digital-to-analog converter (DAC), HOLD buffer, and clock phase generation logic, are presented.

Basic functionality expected from each of these sub-blocks are as follows:

1) Clock Path: Proper design of clock path is very critical for keeping the timing errors low. The clock path mainly comprises of the clock phase generator and circuit for tuning the sampling clock edges. The goal of designing the clock path is to achieve a shortest possible path from actual clock source to actual sampling switch. In other words the number of logic gates between a clock source and a sampling switch should be reduced as much as possible. The clock phase generator creates all the relevant clock phases like masking clock phases, bottom-plate sampling clock and HOLD phases.

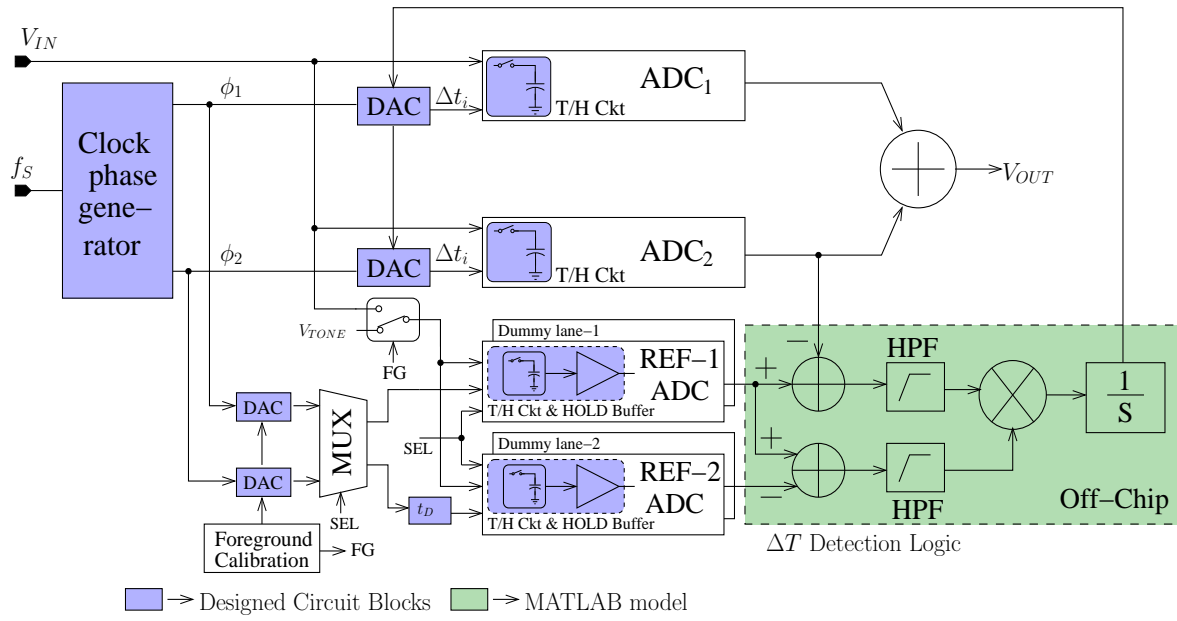


Figure 5-1: System-level block diagram of timing error calibration loop.

2) Track-and-Hold: For this design, the input frequencies up till 1GHz would be supported. For these high input frequencies the track-and-hold should be sufficiently linear. Also, its sizing should be such that it can be sliced into 16x smaller version to implement a reference lane and a dummy sampling lane. The scaling should be precise to keep the bandwidth of main-lane and reference/dummy lane sampling circuit, exactly the same.

3) HOLD buffer: The sampling capacitor in the sampling front-end of the main-lane and the reference lane is actually a DAC for SAR ADC. After sampling input across it, the charge is redistributed to quantize the input signal. However, in reference lane, as the sampling capacitor is 16x smaller, implementing a DAC becomes quite challenging. A solution around this problem is to employ a HOLD buffer which would transfer the charge from a smaller sampling capacitor to a bigger one.

4) DAC: Timing errors are corrected by tuning the sampling clock edge based on digital correction word calculated by the calibration loop. To achieve this tuning a digital word needs to be converted into an analog delay which is done by first converting digital word into analog voltage and later, this analog voltage is converted into delay. The DAC thus, will convert the digital word into an analog voltage. For achieving correction accuracy of 5fs over a dynamic range of even 5ps would require a high-resolution DAC (e.g. 10bits). Further, as shown in fig.(5-1), two DACs are required per main-lane ADC. Hence, they should be low area and low power.

The sampling clocks for the reference lane are multiplexed. The addition of *MUX* will also hamper timing detection accuracy. Before using the reference lane for timing error detection, it is thus, necessary to bring the sampling clock edges of reference lanes on grid with that of the main-lane. Further, the input-offset and gain mismatch (discussed in Chapter-3) will severely hamper convergence of calibration loop. These problems can be addressed by employing foreground calibration for the reference lane. During foreground calibration the FG signal is asserted and a test signal, V_{TONE} is fed into the reference lane. The foreground calibration logic (implemented off-chip) tunes the DAC input to bring the sampling clocks on the grid with the main-lanes. Input-offset and gain mismatch can be completely corrected in the digital domain.

5-2 Clock Path

As 2x interleaved ADC is assumed for this work, the clock path would mainly consist of a clock-phase generator and a sampling-edge tuning circuit.

5-2-1 Clock-Phase generator

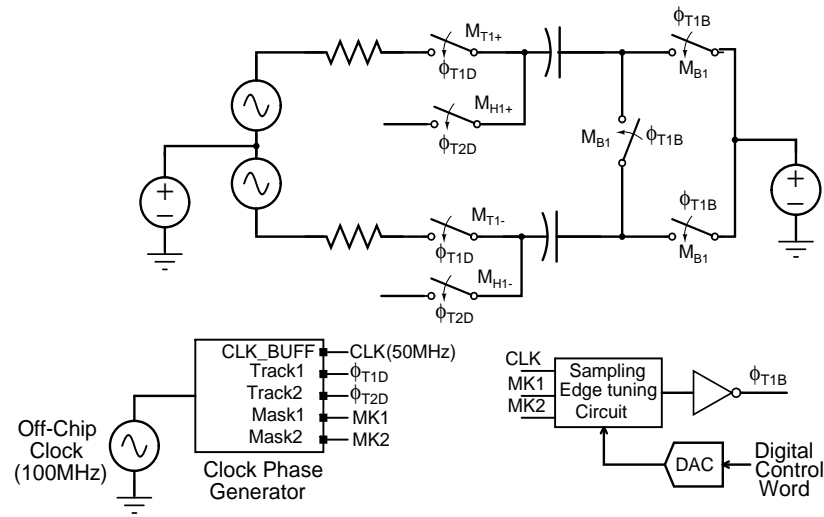
The main job of this block is to generate all the relevant timing phases for the two interleaved main-lane ADCs. Typically, the interleaving of two ADCs is realized by interleaving their clock phases. These clock phases may or may not be timing critical. For instance, the clock phase that drives the bottom-plate switches in fig.(5-2)(a) (marked M_{B1}) is extremely timing critical signal as it determines the sampling instances of the lane-ADC. If the timing critical signals are interleaved then the spread in delay of digital logic that generates these signals will also add into the timing errors. In order to avoid this, as shown in [19], masking clocks are interleaved which are not timing critical. These masking clocks act as a filter which selects a relevant clock-edge and generates the sampling-edge, ϕ_{TB} , for each lane-ADC. For example, as shown in fig.(5-2)(b), when the Mask-1 signal is high, the clock edge marked as Lane-1 generates the sampling edge ϕ_{T1B} for lane-1. Similarly, Mask-2 signal generates sampling-edge ϕ_{T2B} for lane-2 by masking respective clock-edge. As the bottom-plate switches are PMOS transistors, the sampling instances are determined by the rising edge of ϕ_{TB} . The rise and fall times of the masking clock is not critical as they are merely acting as a filter which selects a relevant clock-edge.

The track phases that drive the top plate switches, M_{T1+} and M_{T1-} , (fig.(5-2)(a)) are also generated by the clock phase generator. During a track phase the sampling front-end tracks the input signal. For 2x interleaved ADC, the two track-phases can be reused as hold phases as long as they are non-overlapping. Thus, the track phase ϕ_{T1D} of lane-1 can be used as the hold phase for lane-2. Similarly, ϕ_{T2D} can be used as the hold phase for lane-1 while it serves as track phase for lane-2. A summary of clock phases that are created by the clock phase generator is given below:

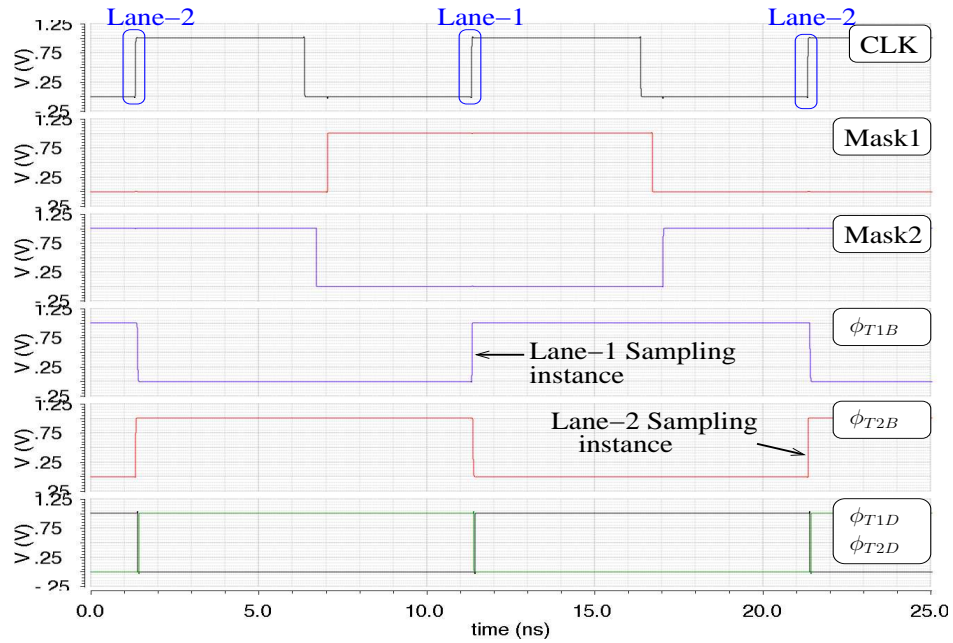
- 1) $\phi_{T1D} \Rightarrow$ Tracking phase of lane-1. It is delayed version of actual track-phase. Delayed to realize bottom-plate sampling. It is non-overlapping with the tracking-phase of lane-2.
- 2) $\phi_{H1} \Rightarrow$ Hold phase of lane-1. For 2x interleaving it is equal to track-phase of lane-2 i.e. ϕ_{T2D} .
- 3) $\phi_{T1B} \Rightarrow$ Bottom-plate sampling phase of lane-1. Its rising-edge samples the input signal.
- 4) $\phi_{T2D} \Rightarrow$ Tracking phase of lane-2. It is delayed version of actual track-phase. Delayed to implement bottom-plate sampling. It is non-overlapping with the tracking-phase of lane-1.
- 5) $\phi_{H2} \Rightarrow$ Hold phase of lane-2. For 2x interleaving it is equal to track-phase of lane-1 i.e. ϕ_{T1D} .
- 6) $\phi_{T2B} \Rightarrow$ Bottom-plate sampling phase of lane-2. Its rising-edge samples the input signal.

5-2-2 Sampling Edge Tuning Circuit

Fig.(5-3)(a) shows the complete circuit diagram of the main-lane clock path. Depending upon the masking signal the respective clock-edge is allowed to sample the input. It might be apparent that



(a) Circuit diagram demonstrating various phases of sampling front-end for lane-1.



(b) Clock phases for both lane-1 and lane-2 sampling front-ends.

Figure 5-2: Block diagram of clock path.

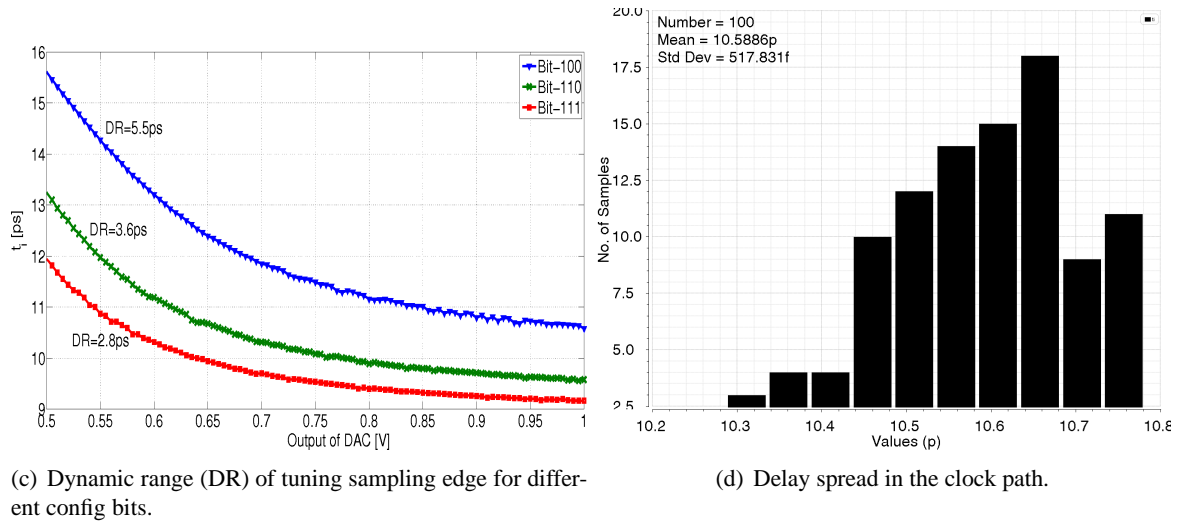
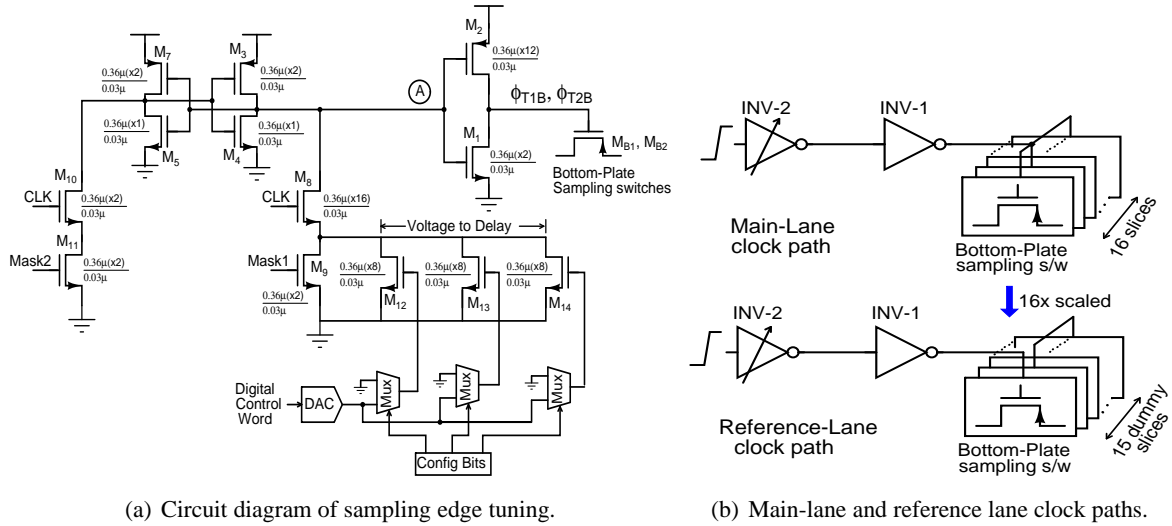


Figure 5-3: Sampling edge tuning circuit and its simulation results.

the scheme shown in fig.(5-3)(a) forms the shortest path from the actual clock edge to the sampling switch. Thus, the only elements that add intrinsic delay to the clock path are M_8 , M_1 and M_2 (one transistor and one inverter). M_9 is controlled by the masking clock which selects the clock edge that can pull-down the gate of $M_1 - M_2$ inverter thereby opening the bottom-plate sampling switches. However, once the masking phase and the clock edge passes away, the node \textcircled{A} may get pulled-up again as the voltage at this node is floating. If this node gets pulled-up then the bottom-plate switches will close while the ADC is still in the conversion mode. This catastrophic event can be avoided by latching the potential at node \textcircled{A} . Thus, the transistors $M_3 - M_7$ are arranged to implement this latch. The latch is reset by the masking pulse of lane-2.

Transistors $M_{12} - M_{14}$ implement a voltage to delay converter which converts the output voltage of the DAC into delay. These transistors shunt M_9 and control its ON-resistance. The potential at node \textcircled{A} is discharged through M_9 . Controlling its ON-resistance would control the discharge rate and thus, the delay of the sampling edge. Transistors $M_{12} - M_{14}$ also control the dynamic range of the delay correction. Fig.(5-3)(c) shows the variation in the dynamic range for different settings of the

configuration bits.

Further, as suggested in chapter-4, the sampling front end of the reference lane should be at least 16x smaller than the main-lane. Thus, the bottom-plate switch for the reference lane is made 16x smaller compared with the main-lane. Merely slicing the bottom-plate switch of the main-lane to 16x smaller, would result into systematic offset in the timing error as the loading of the two clock-paths would change. The scaling should be done in such a way that the load on $INV-1$ stays the same. Alternatively, complete clock path including the DAC needs to be scaled which is considerably challenging. The loading on $INV-1$ can be maintained by inserting dummy bottom-plate switches which merely load $INV-1$ but do not participate in the sampling process. Thus, only the size of the bottom-plate sampling switches M_{B1} (fig.(5-2)(a)) is scaled by 16x and the size of the inverters $INV-1$ and $INV-2$ stays unscaled.

The spread in the delay of the clock path will be calibrated by the calibration loop. Fig.(5-3)(d) shows that the clock path has a mean delay of about $11ps$ which spreads by $0.16ps$. The calibration loop should be able to correct this spread in delay of the clock path to 3σ level. Thus, the variance of the clock path σ_{CLK} that needs to be considered is $0.5ps$. This value will be later used to estimate the dynamic range of the DAC.

5-3 Track-and-hold Design

The track-and-hold circuit samples the input signal on the sampling capacitor C_1 . This sampled voltage is later converted by the ADC into a digital code. For the present design, the maximum input frequency of 1GHz needs to be supported. At such high frequencies designing a track-and-hold with high linearity is challenging. However, as the focus of this thesis has been on calibration of timing errors, the linearity should be good enough to allow identification of small timing error spurs from the harmonic spurs. For this reason the target HD_3 specification is around 70dB instead of 103dB (refer Appendix-A). Thus, the single most important design metric for the track-and-hold circuit is its bandwidth. As the maximum input frequency of 1GHz needs to be supported, the bandwidth of the track-and-hold should be at least 5GHz to minimize timing errors due to bandwidth mismatch (Section-2.2.4, Chapter-2).

Fig.(5-4)(a) shows the circuit diagram of the fully-differential track-and-hold. There are two track-and-hold circuit shown, one is for the main-lane ADC and the other one is for the reference lane ADC. As will be shown later, the track-and-hold circuit for the reference lane is 16x smaller compared to that of the main-lane (refer Chapter-4). Every element used in the track-and-hold for main-lane is thus, 16x smaller for reference lane including bootstrap circuit, sampling capacitor and all the switches. The input buffer is shared between main-lane and reference lane and so, its output impedance represented by R_S cannot be scaled. For simulation purposes, its value is chosen to be 20Ω . The operation of the track-and-hold circuit is divided into a TRACK phase and a HOLD phase. These phases are shown in fig.(5-2)(b). For the sake of argument, the clock phases shown in fig.(5-4)(a) correspond to lane-1 ADC. Hence, the HOLD phase, ϕ_{H1} would be actually the TRACK phase of lane-2.

To better understand the functioning of this circuit consider the main-lane track-and-hold circuit alone. During the TRACK phase the voltage across the sampling capacitor follows the input signal. The top-plate switches M_{T1+} and M_{T1-} together with the bottom-plate switches M_{B1} are closed. The bandwidth of the track-and-hold is determined by the tracking time-constant. This time-constant

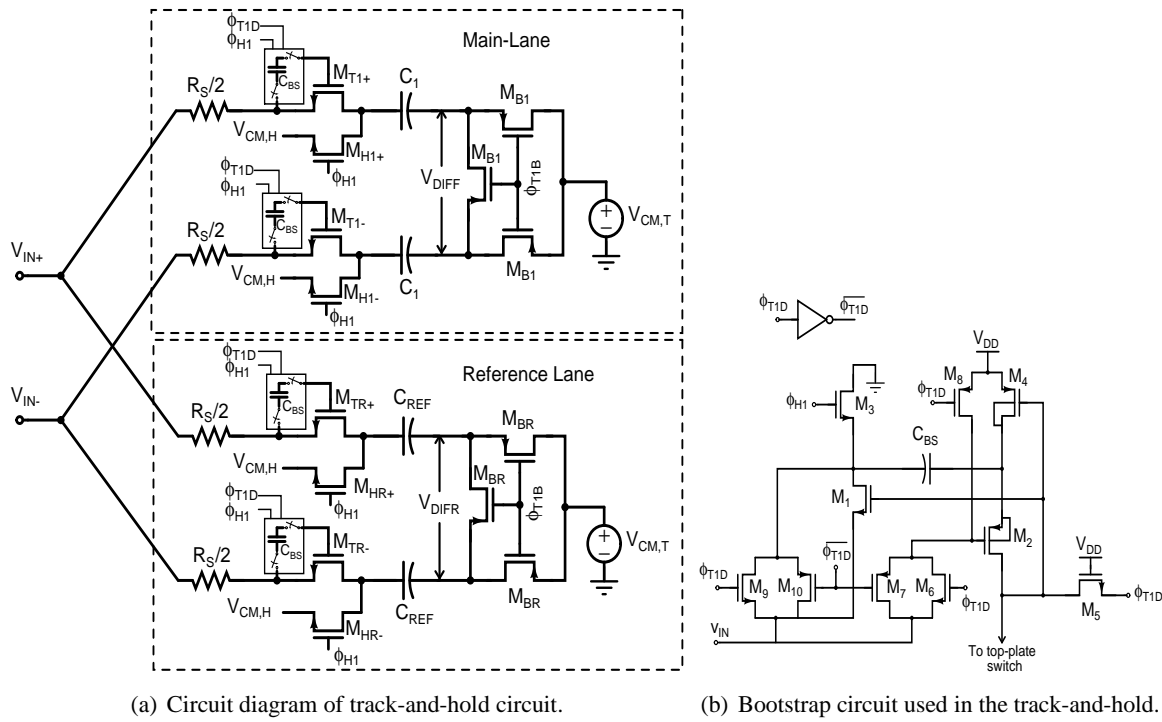
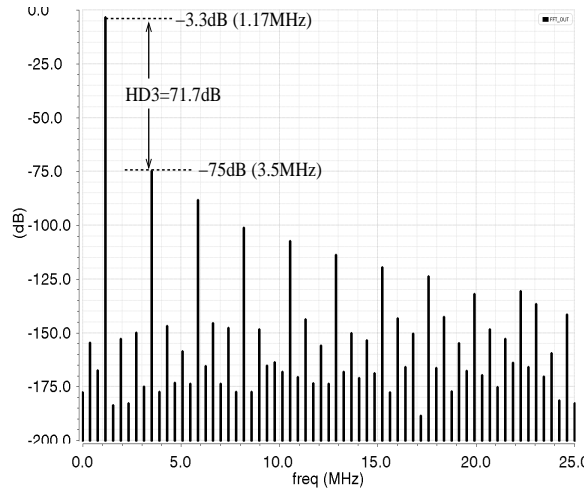
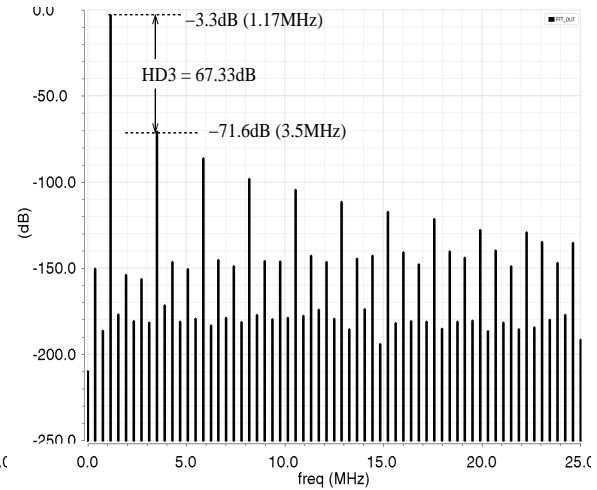
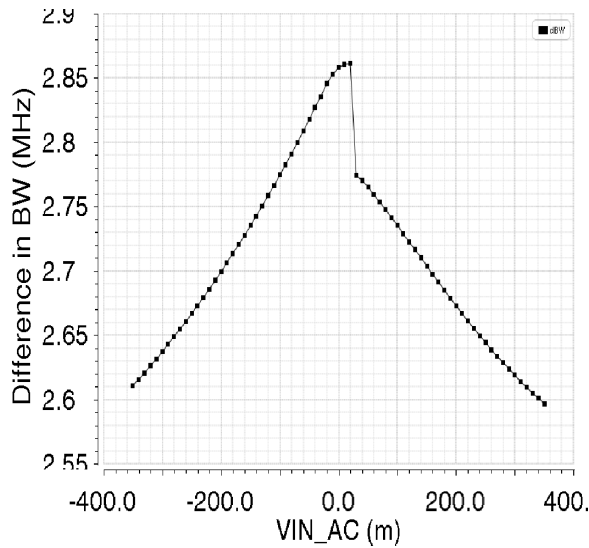


Figure 5-4: Simulation results of the track-and-hold circuit for main-lane and 16x scaled reference lane.

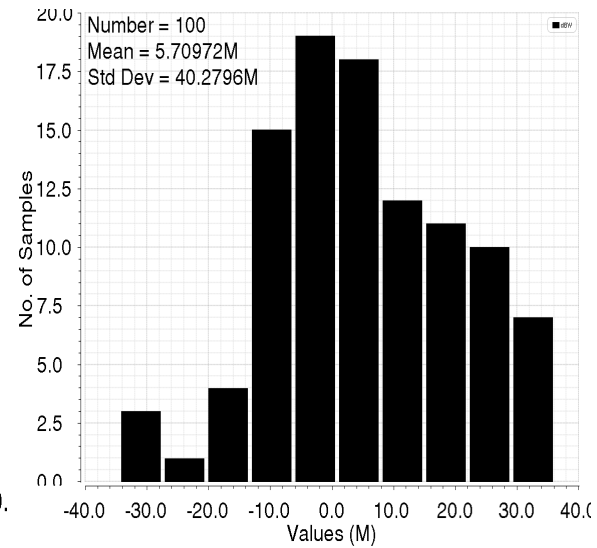
is merely the product of the ON-resistance of the top-plate switch and the sampling capacitor. To achieve higher tracking bandwidth this time-constant needs to be reduced. The size of the sampling capacitor is governed by the target noise specifications and it cannot be reduced. However the ON-resistance of the top-plate switch can be reduced by employing the bootstrap circuit. Fig.(5-4)(b) shows the transistor level diagram of bootstrap circuit. In addition to reducing the ON-resistance, the bootstrap circuit also makes it constant with the input signal, thereby enhancing the linearity of the track-and-hold. Also, the arrangement of the bottom-plate switches is such that their ON-resistance do not increase the tracking time-constant.

After the TRACK phase is over, the input signal is sampled across C_1 by first opening the bottom-plate switches M_{B1} and after some delay, opening the top-plate switches. Thus, the principle of bottom-plate sampling is realized. This is essential because to minimize tracking time-constant the top-plate switches are made quite large. These large switches inject significant amount of input signal dependent channel charge on C_1 degrading the linearity of the track-and-hold. The output voltage V_{DIFF} is defined during the HOLD phase i.e. when the common-mode switches M_{H1+} and M_{H1-} turn-ON. $V_{CM,T}$ and $V_{CM,H}$ are the common-mode voltages during the TRACK and HOLD phases respectively. The topology adopted for the bootstrap circuit is similar to the one shown in [34]. The sizes of all the devices used in the track-and-hold circuit of the main-lane and the reference lane (scaled by 16x) is given in table-5-3 whereas, the device sizes for the bootstrap circuit is given in table-5-3.

Fig.(5-5) shows the simulation result for the track-and-hold. For 1GHz input sampled at 50MHz, fig.(5-5)(a) shows the FFT plot. It is evident that the HD_3 of the main-lane track-and-hold is about 72dB. This linearity is fair enough to allow identification of spurs due to timing errors. Fig.(5-5)(b)

(a) FFT of V_{DIFF} for 1GHz input signal sampled at 50MHz(b) FFT of V_{DFR} 

(c) Difference in bandwidth of main-lane and scaled reference lane.



(d) Spread in the bandwidth difference.

Figure 5-5: Simulation results of the track-and-hold circuit for main-lane and 16x scaled reference lane.

Main-Lane		Reference Lane (16x Smaller)	
Element	Size ¹	Element	Size ²
M_{T1+} M_{T1-}	96	M_{TR+} M_{TR-}	6
M_{H1+} M_{H1-}	48	M_{HR+} M_{HR-}	3
M_{B1}	16	M_{BR}	1
C_1	1.4pF	C_{REF}	87.5pF
$V_{CM,T}$	0.8V	$V_{CM,T}$	0.8V
$V_{CM,H}$	0.15V	$V_{CM,H}$	0.15V

Table 5-1: Device sizes for the track-and-hold circuit.

[^{1,2} Multiple of unit size 1.28 μ m/0.03 μ m.]

Main-Lane		Reference Lane (16x Smaller)	
Element	Size	Element	Size
M_1, M_2	$(\frac{1.0\mu}{0.03\mu}) \times 16$	M_1, M_2	$(\frac{1.0\mu}{0.03\mu})$
M_3, M_7	$(\frac{0.5\mu}{0.03\mu}) \times 16$	M_3, M_7	$(\frac{0.5\mu}{0.03\mu})$
M_4, M_6 M_8, M_9	$(\frac{0.25\mu}{0.03\mu}) \times 16$	M_4, M_6 M_8, M_9	$(\frac{0.25\mu}{0.03\mu})$
M_5	$(\frac{1.0\mu}{0.03\mu}) \times 32$	M_5	$(\frac{1.0\mu}{0.03\mu}) \times 2$
M_{10}	$(\frac{0.125\mu}{0.03\mu}) \times 16$	M_{10}	$(\frac{0.125\mu}{0.03\mu})$

Table 5-2: Device sizes for the boot-strap circuit

shows the FFT plot for the output of reference lane. The reference lane is scaled 16x smaller than the main-lane still the HD_3 performance is quite on par with that of the main-lane. While scaling the track-and-hold for the reference lane it should be ensured that the bandwidth of both the reference lane and the main-lane is still equal. If this condition is not meet then an additional source of timing errors would stem from it. Fig.(5-5)(c) shows the difference in bandwidth ranges from 2.6MHz to 2.85MHz for input amplitude spanning from +350mV to -350mV.

The spread in the difference of bandwidths is 40.3MHz. The simulated bandwidth of track-and-hold for the main-lane ADC is 5.2GHz. Thus, the spread in a bandwidth is about 0.7%. A bandwidth of 5.2GHz corresponds to tracking time-constant of 32ps. Thus, as spread in bandwidth of 0.7% would mean a spread in timing error of 0.224ps (0.007*32ps). The $3 \cdot \sigma$ value for the spread in difference of bandwidth, σ_{BW} thus, equals 0.8ps. This value will be used later while estimating the dynamic range of the DAC.

5-4 HOLD Buffer Design

As the sampling capacitor of the reference lane is only 87.5fF, it is challenging to perform charge redistribution using such a small capacitance. Alternatively, a HOLD buffer can be used, as shown in fig.(5-6)(a), to move the sampled input signal over a larger capacitor of $REF-ADC$. To implement this buffering functionality a fully-differential folded-cascode OTA with switch-capacitor CMFB is chosen. Fig.(5-6)(a) shows the circuit diagram of the OTA along with its respective sizes. The input pair $M_3 - M_4$ achieves a transconductance of 3mS when biased by 128 μ A current. Transistors M_5 and M_6 are used to isolate the output impedance of the input pair from the folding node (drain of $M_7 - M_{10}$). This helps to achieve slightly higher loop-gain. The output common-mode is set by using a switch capacitor based common-mode feedback (CMFB) as shown in fig.(5-6)(c). The correction voltage V_{CMFB} is applied to the gate of $M_7 - M_{10}$. The value of V_{CMFB} is adjusted to correct for the error between required output common-mode V_{CM} and ideal bias voltage V_{BIAS} . Capacitor C_2 samples this difference in voltage during ϕ_T and copies this voltage across C_1 during ϕ_H . To generate V_{BIAS} a replica circuit as shown in fig.(5-6)(c) is used. The switches are sized to settle to the error voltage within 10ns while minimizing their charge injection. Fig.(5-6)(d) shows the loop-gain

Table 5-3: Specifications achieved by the HOLD buffer design.

Parameter	Value	Remark
Sampling Speed	50MHz	10ns for charge transfer
Output Common Mode	0.65 V	Required by <i>REF</i> ADC
Output Swing	1.34V _{PP}	Swing to be supported by ADC
C_L and g_m	1.4pF and 3mS	<i>REF</i> ADC sampling cap
Loop Gain and UGB	60dB (at DC) and 250MHz	
Phase Margin	78°	
Input-offset	4.8mV	Monte-Carlo sim over 200 runs
Total Power	1.7mW	Including bias circuit

magnitude and the phase plot. The HOLD buffer has 61dB gain at DC and 78° phase margin at UGB of 250MHz.

Fig.(5-7) shows the transient simulation results of the HOLD buffer. The output signal swing is about 1.34V peak-peak as seen in fig.(5-7)(a). The deviation of 66mV from the target value of 1.4V_{PP} is due to less than 20dB loop gain (fig.(5-6)(d)) at 50MHz sampling speed. Fig.(5-7)(b) shows the third harmonic distortion of about 40dB. Note that the main tone is located at 1.17MHz because the input frequency of 998828125 Hz (≈ 1 GHz) is sampled at 50MHz. Further, the input offset of the HOLD buffer is simulated across 200 Monte-Carlo runs in the presence of mismatch (fig.(5-7)(c)). The distortion and the input-offset are the non-idealities of HOLD buffer that may hamper the calibration loop performance. In order to investigate that effect, the input-offset and distortion of the HOLD buffer are modeled in the MATLAB model of calibration loop. Fig.(5-7)(d) shows that the loop might had convergence problems if there was no high-pass filter (HPF) in the loop. The distortion of the buffer would generate a DC component which would be integrated by loop causing it to converge to a fall steady-state value. However, as a high-pass filter is employed the DC-component generated due to non-idealities of the buffer are blocked from appearing at the input of the integrator. The second plot in fig.(5-7)(d) confirms this assertion. Table-5-3 summarizes important design specifications met by the HOLD buffer design.

5-5 Digital-to-Analog Converter (DAC)

This section details the design of the DAC that converts the control word feedback by the calibration loop, into an analog voltage. Later, this analog voltage is converted into delay which tunes the sampling clock instance. As the calibration loop requires two DACs for each main-lane ADC, the area and power overhead of the DAC should be as low as possible by design. Also, the timing error needs to be corrected down to 5fs level which requires DAC to be sufficiently accurate. Thus, the design of DAC has three main metrics namely, DNL (a measure of accuracy), area and power.

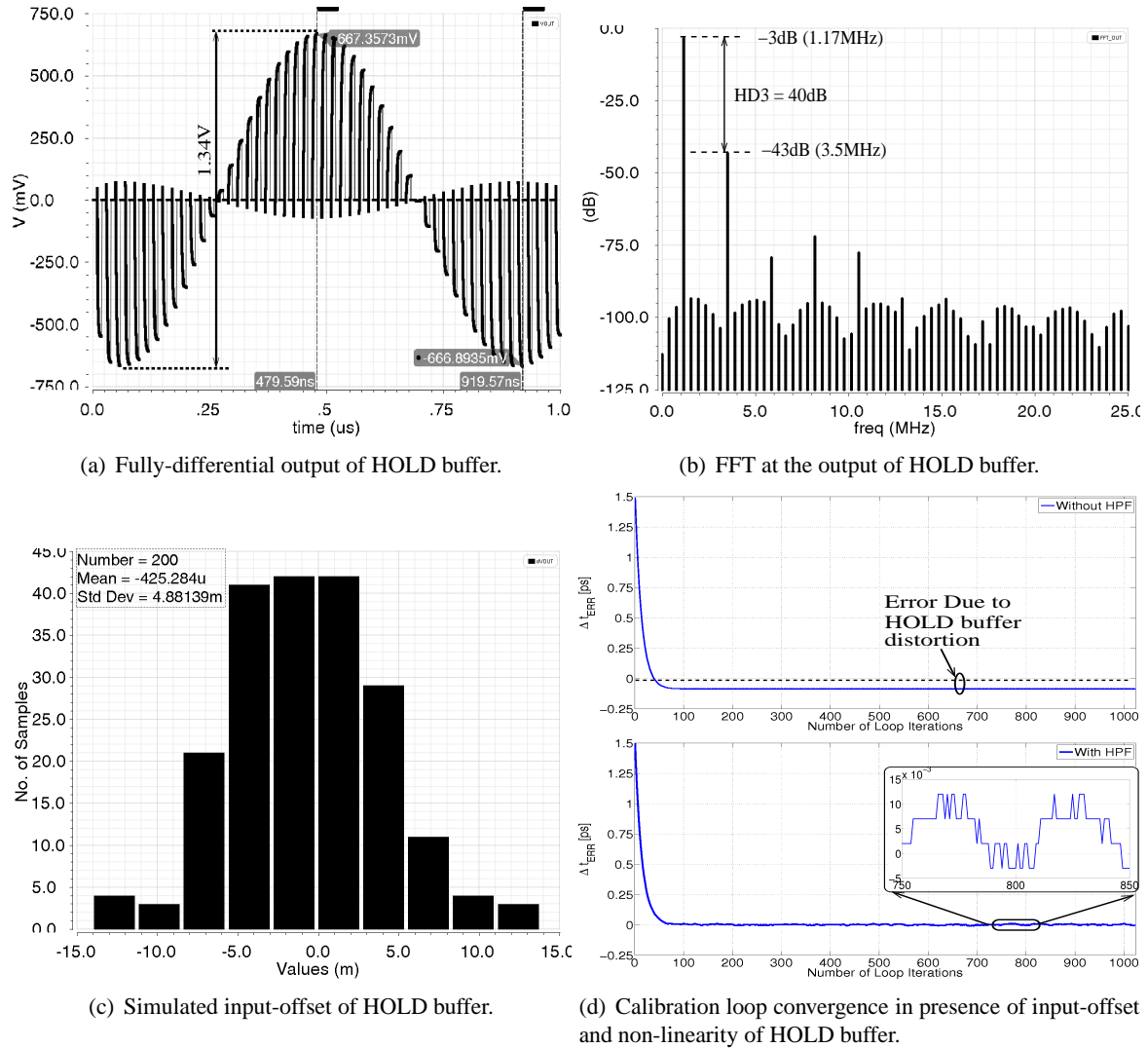


Figure 5-7: Simulation Results of HOLD buffer.

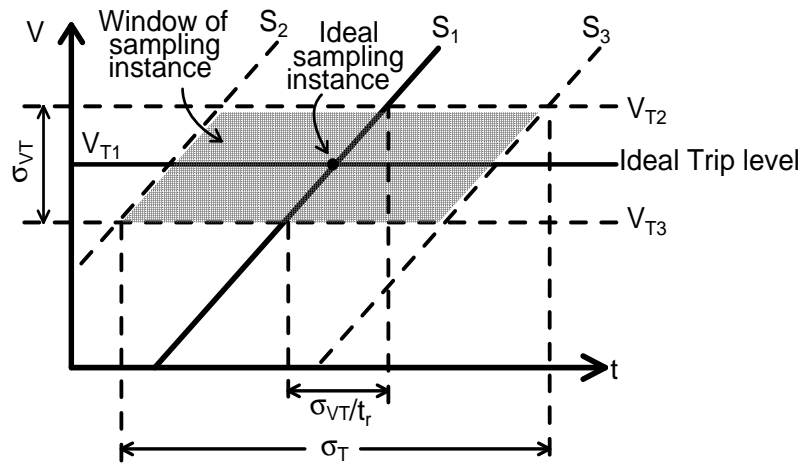


Figure 5-8: Window of sampling instance due to various sources of timing errors.

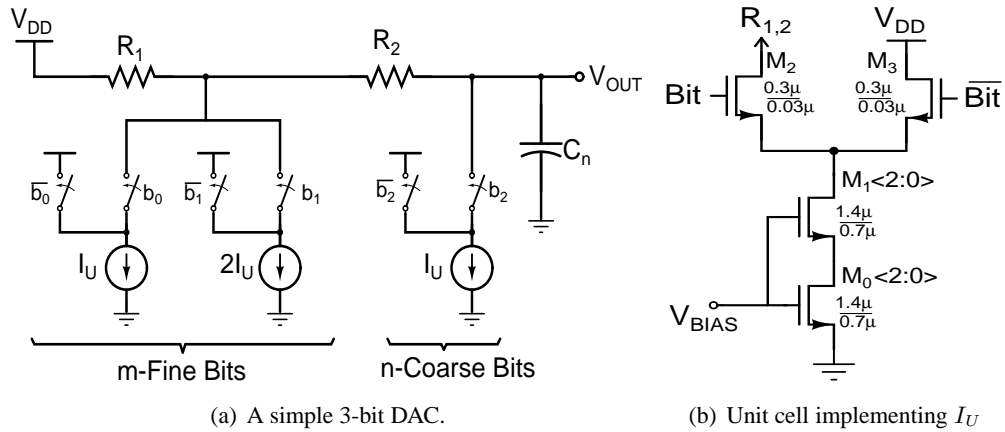


Figure 5-9: Fine-coarse arrangement and unit cell of the DAC.

5-5-1 Estimating Dynamic Range

Before embarking on the DAC design, it is vital to get an estimate of the expected dynamic range wide enough for the DAC to compensate for all possible sources of timing errors. In chapter-2, sources of timing errors for passive track-and-hold were presented which can be categorized into errors due to bandwidth mismatch and clock path mismatch. In the sections on the track-and-hold design and the clock-path design, the simulation results for these two sources of timing error were presented.

Under ideal condition, as shown in fig.(5-8), the sampling instance is a point defined by the intersection of the sampling edge and the ideal trip level V_{T1} (or threshold level). However, due to spread in bandwidth and clock-path, the sampling edge S_1 can be anywhere between S_2 and S_3 . Similarly, the threshold level can spread anywhere between V_{T2} to V_{T3} . This creates a window within which the sampling instance would fall. Note that it is not important for the sampling instance to be in the ideal position rather the sampling instances of all main-lane should be aligned with each other.

The total $6 \cdot \sigma$ spread in timing error, σ_T is shown pictorial by fig.(5-8). It can be expressed mathematically as,

$$\sigma_T = 3 \cdot \sqrt{\sigma_{BW}^2 + \sigma_{CLK}^2 + \sigma_{ADL}^2 + \left(\frac{\sigma_{VT}}{t_r}\right)^2} \quad (5-1)$$

Note that the variance that are added in RMS are already $3 \cdot \sigma$ values. From previous sections, it is known that, the spread in timing error due to bandwidth mismatch, σ_{BW} is 0.8ps while that from the clock path, σ_{CLK} is 0.5ps. The spread in threshold voltage, σ_{VT} is around 3.3mV and for a rise time of about 3ps, the timing error stemming from it is roughly 0.66fs. Such a small amount of timing error can be safely ignored. Lastly, timing error due to spread in delay line σ_{ADL} can also be ignored as the delay lines are made from metal wires which have negligible spread due to its larger width and length. Additionally, the mean of the delay added by the delay line itself is in the order of few pico-seconds. Thus, using the bandwidth mismatch and clock path mismatch in eqn.(5-1), the total spread can be obtained as,

$$\sigma_T = 2.8ps \approx 3ps \quad (5-2)$$

Thus, the dynamic range of the DAC should be at least 3ps or more. With step-size of 5fs, this results in minimum DAC resolution requirement of 10 bits.

5-5-2 Circuit Implementation

Next, an attempt is made to design a low-power and low-area DAC with 11 bit resolution (mandatory 10 bits + 1 bit over-range). There are several choices available while looking for right DAC topology. In the present design a current steering, coarse-fine topology is chosen where 3 fine bits are binary and 8 coarse bits are thermometer.

To better understand the functioning of the DAC, consider a schematic shown in fig.(5-9)(a) where a simple 3-bit DAC is realized with 2 binary fine-bits (i.e. $m = 2$) and 1 thermometer coarse-bit (i.e. $n = 1$). As it appears, this DAC topology is actually a cascade connection of two current-steering DAC with I_U as a unit current source element. The circuit implementation of the unit cell is shown in fig.(5-9)(b). When both currents are steered to supply ($V_{DD} = 1.0V$) the output voltage, V_{OUT} is zero. The minimum value of V_{OUT} is 1 LSB which is generated by turning-on one I_U in the fine-bits bank. Thus $I_U \cdot R_1$ is the LSB of the DAC. When the coarse current source I_U is turned on, the V_{OUT} is $(R_1 + R_2) \cdot I_U$. The relationship between R_1 and R_2 should be such that every time a coarse current

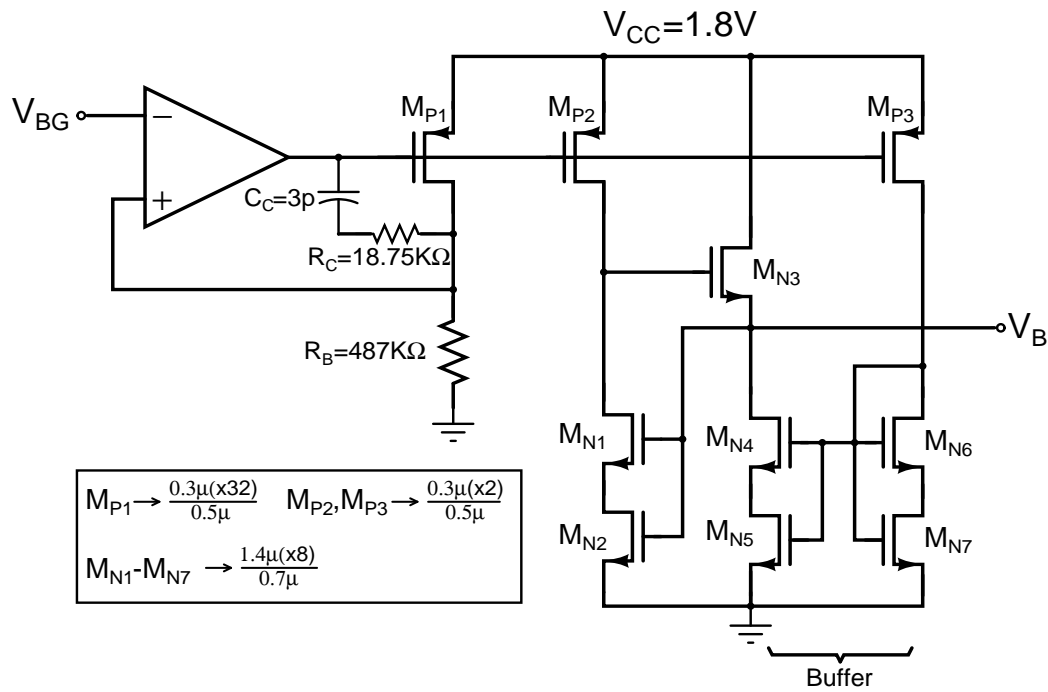
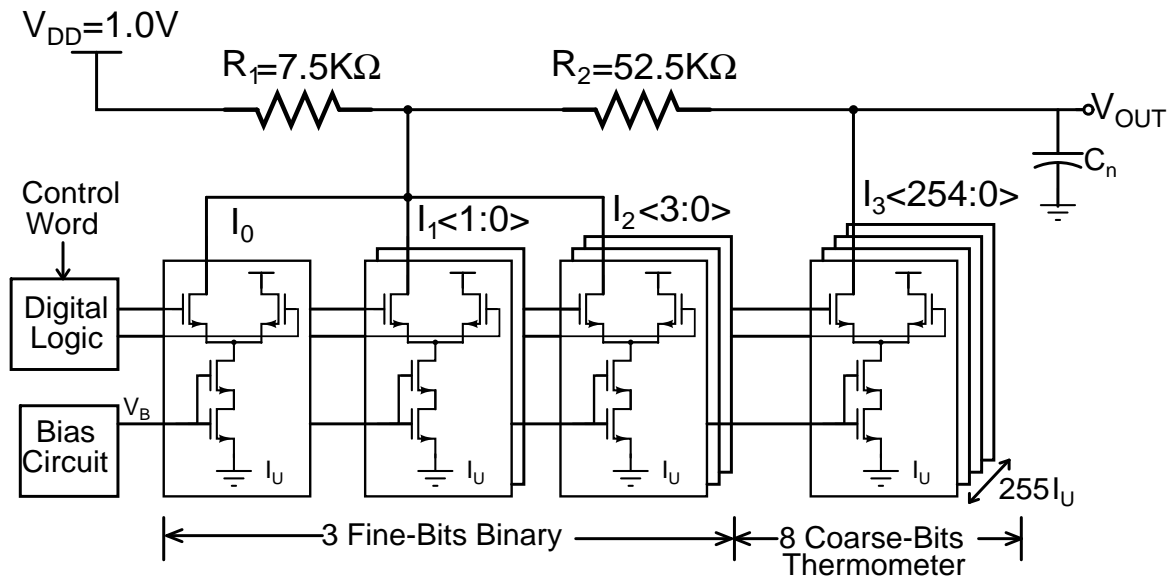


Figure 5-10: Circuit-level implementation of 11-bit DAC.

source is turned on, a jump of $(1LSB + 7 \cdot I_U R_1)$ is made. Expressing this mathematically, the coarse step should satisfy,

$$I_U \cdot (R_1 + R_2) = \underbrace{I_U R_1}_{1 \text{ LSB}} + 7 \cdot I_U R_1 \quad (5-3)$$

$$\therefore R_2 = 7 \cdot R_1 \quad (5-4)$$

This expression can be extended and a more generalized relationship between R_1 and R_2 can be written as,

$$R_1 = \frac{R_2}{(2^m - 1)} \quad (5-5)$$

where, m is the number of fine-bits. Also, the output voltage, V_{OUT} , of such a N-bit DAC can be written as,

$$V_{OUT} = V_{DD} - I_U \cdot R_1 \cdot (b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \dots + b_0 \cdot 2^0) \quad (5-6)$$

where, $N (=m+n)$ is the total number of bits.

Using this basic set of equations a 11-bit DAC is designed as shown in fig.(5-10)(a). The unit current source I_U is implemented as shown in fig.(5-9)(b). In order to have a high output impedance, NMOS transistor M_1 is placed on top of M_0 to increase their effective channel length. Due to process limitations the channel length of M_1 can be maximum of $1\mu m$ long. As low-power is an important specification, one obvious technique would be to shut-off the current source I_U instead of steering it V_{DD} . The problem is as the output impedance of the switch M_2 in off-state is finite, a small drain-source leakage will still exist. By adding a current steering switch M_3 , the source of M_2 is pulled higher which makes its V_{GS} negative instead of being zero. Hence, M_2 closes firmly which significantly reduces the drain-source leakage. The sizes of switches M_2 and M_3 are minimized in order to reduce the drain-bulk leakage. Particularly, at the drain node of M_2 , many other switches from adjoining unit cells would be connected, the drain-bulk leakage will be significant. Further, this leakage component flows through R_1 and R_2 creating an error voltage. Note that this error voltage change with the input code and will degrade the INL of the DAC. As the DAC is used in the feedback in conjunction with the calibration loop, its INL is not important. But from a power point of view, it is desirable to minimize the drain-bulk leakage of both M_2 and M_3 .

This unit cell is used to build 3 binary fine-bits and 8 thermometric coarse-bits in a 11-bit DAC (fig.(5-10)(a)). The values of important elements are: $V_{DD} = 1.0V$, $R_1 = 7.5K\Omega$, $R_2 = 52.5K\Omega$ and $I_U = 32nA$. The dynamic range of the DAC is from 0.5V to 1.0V with step size of $240\mu V$. To achieve power consumption of around $10\mu W$, the value of I_U was chosen to be 32nA. For this value of I_U and one LSB of $240\mu V$, the value of R_1 is $7.5K\Omega$. Using eqn.(5-5) for $m=3$ bits, results in the value of $R_2 = 52.5K\Omega$.

The DAC is targeted only for DC performance as the update rate of the calibration loop is slow. In such case the noise requirement can be simply met by shunting the V_{OUT} node with a capacitor C_n ($3pF$) for filtering the thermal noise. The digital logic generates required control signals for both binary and thermometric arrangement of the unit current sources. The digital logic is controlled by the calibration loop through a digital control word which is 11-bits wide. Fig.(5-10)(b) shows the bias generation circuit that generates bias voltage V_B for all the unit current sources of the DAC. The PMOS transistor M_{P1} carries a current equal to V_{BG}/R_B . This current is scaled down $16x$ times

to bias M_{N1} and M_{N2} . As the bias node V_B has a load of about 262 current sources, it needs to supply sufficient gate leakage current of these current sources. This gate leakage current should not flow through M_{P2} (fig.(5-10))(b) otherwise the mirror ratio will be altered. The way around this problem is to add a buffer transistor M_{N3} which would supply the necessary gate leakage current. The buffer is biased by M_{N4} and M_{N5} . The size of all PMOS and NMOS transistors is shown in fig.(5-10)(b). Note that R_B in the bias circuit and R_1, R_2 in the DAC, have to match with sufficient precision. Hence, all the resistors are made out of unit resistor of value $15K\Omega$ which has a length of $12.5\mu m$ and width of $0.5\mu m$.

5-5-3 Simulation of DNL

The LSB of the DAC is $240\mu V$. To achieve a DNL specification of less than an LSB over 3σ , it should satisfy,

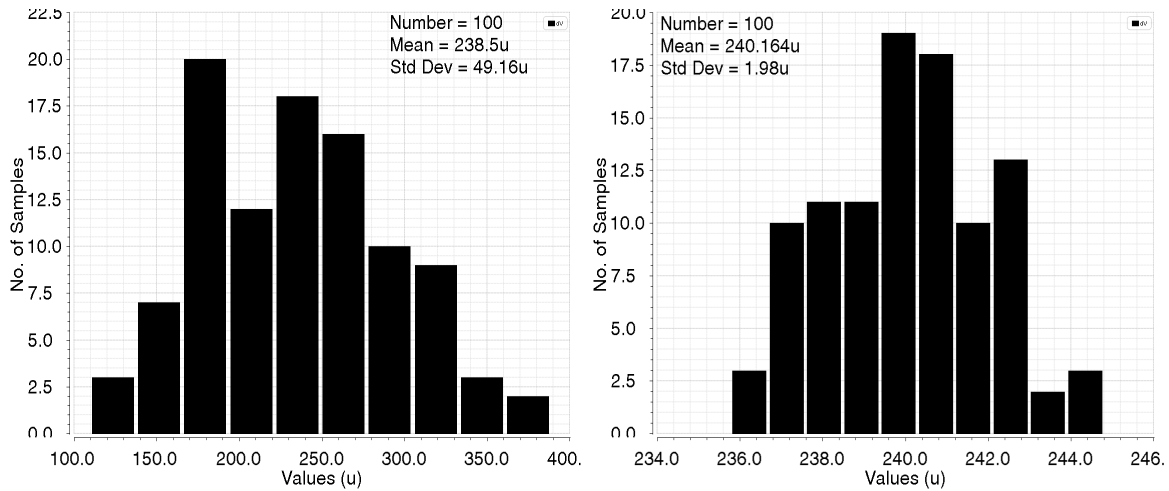
$$\sigma_{DNL} \leq \frac{1LSB}{6} \approx 40\mu V \quad (5-7)$$

There are two main factors that contribute to σ_{DNL} . First is the mismatch in the unit current sources and second, is the mismatch in the resistors R_1 and R_2 . The value of σ_{DNL} is maximum during the code transition from the fine to the coarse segment. This transition is the major code transition for DAC schematic shown in fig.(5-10)(a). During this transition all the fine-bits are turned-off and one coarse bit is turned-on. The condition of eqn.(5-7) should be fulfilled for this major code transition to achieve less than 1LSB over $3 \cdot \sigma$. Fig.(5-11)(a)-(b) shows the spread in the major code LSB in the presence of a mismatch in current source and resistors, respectively. The size of the current source and the resistor should be adjusted such that the RMS sum of the two variances would be less than or equal to $40\mu V$. For the unit current source sized as shown in fig.(5-9)(b), the spread in major code transition in the presence of current source mismatch, is around $49\mu V$. Whereas, for R_1 and R_2 of width $0.5\mu m$, the spread in major code transition is about $2\mu V$. The RMS sum of the two variances is approximately $49\mu V$. This value is higher than $40\mu V$. One simple way to meet this requirement is to double the width of the unit current source (M_0 and M_1 in fig.(5-10)(b)). This would scale the variance of the major code by a factor of $1/\sqrt{2}$ to $34.64\mu V$. However, it will also double the analog area. In order to keep the analog area of the current sources low, the DNL is compromised and the area of the unit current sources is not doubled. This is a classic area and accuracy trade-off seen in all current steering DACs [35]. Note that if DNL is $> 1LSB$ then the DAC will not be monotonic. However, this is not a deterrent drawback for the calibration loop.

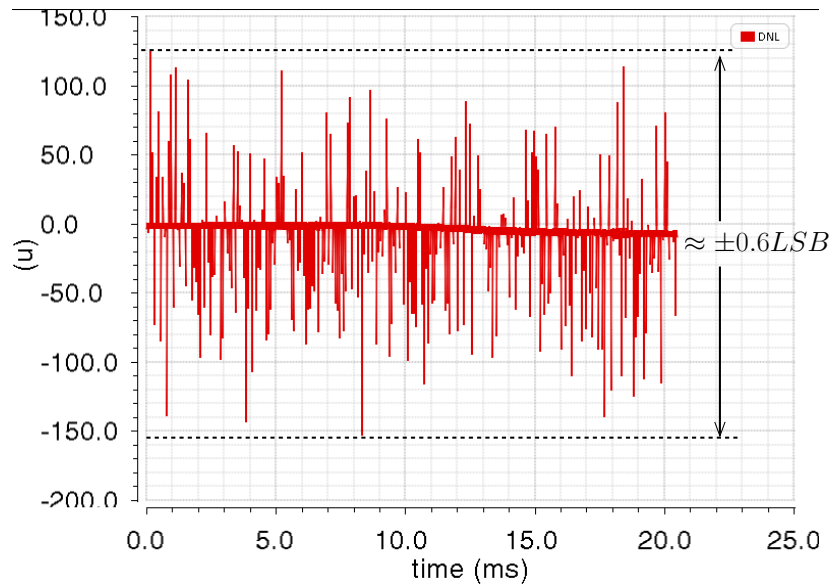
Fig.(5-11)(c) shows the DNL of the complete DAC. Single Monte-Carlo simulation is performed for complete DAC schematic which includes the bias circuit and the digital control logic. It is apparent that the DNL of the DAC is around $0.6LSB$ which is not what was targeted. The difference is because of the area-accuracy trade-off being made to the current source where the variance in the major code LSB due to current source mismatch was increased to $49\mu V$ instead of $40\mu V$.

5-5-4 Design Summary

Accuracy, area and power overhead, are the three vital performance metrics for the design of DAC. To estimate the layout-area of the DAC, both analog and digital layout area should be computed. The analog area occupied by 262 current cells can be estimated from the area occupied by the unit cell. Fig.(5-12)(a) shows the layout of a unit cell whose schematic was shown in fig.(5-9)(b). The



(a) Spread in major code transition due to mismatch only in current sources only. (b) Spread in major code transition due to mismatch only in resistors.



(c) DNL for 1 Monte-Carlo run with complete DAC.

Figure 5-11: DNL simulations of DNL of 11-bit DAC.

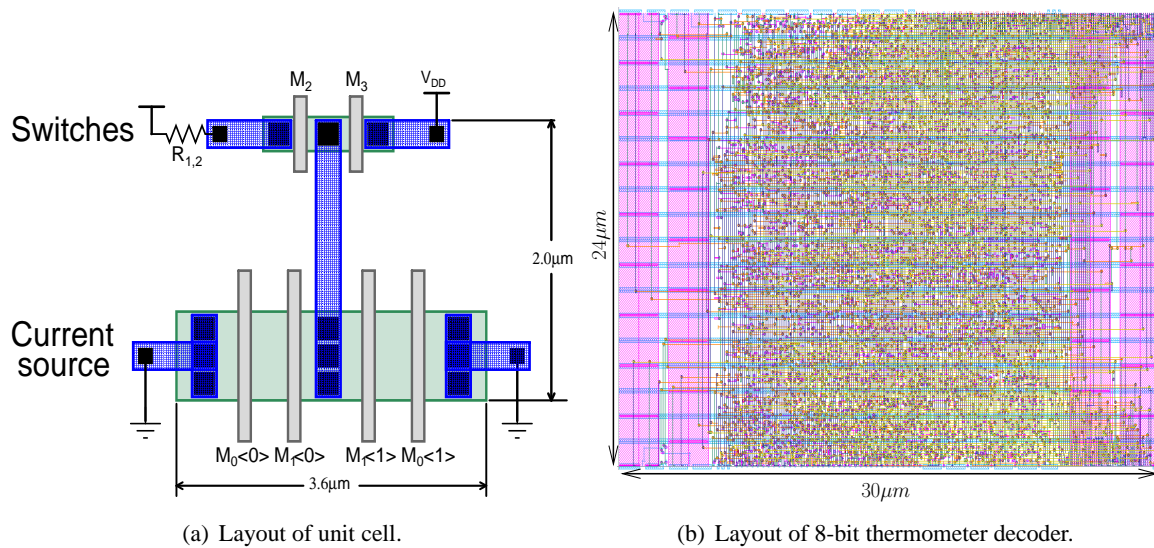


Figure 5-12: Area estimation of 11-bit DAC.

Table 5-4: Specifications achieved by the 11-bit DAC design.

Parameter	Value	Remark
V_{OUT} Dynamic range	0.5 – 1.0V	Required by sampling edge tuning circuit.
DNL	0.6LSB	RMS over 1 Monte-carlo run of complete DAC.
Area	2610 μm^2	Analog + Digital combined, but without routing
Total Power	8.4 μW	Only current sources. Digital power ignored
Integrated Noise	40.3 μV	at 80°

current source has a channel length larger than the minimum. Hence, they can share the same OD layer resulting in a much tighter layout. An area of one current source is around $7.2 \mu m^2$ which scales to $1890 \mu m^2$ for 262 current sources. Note that wire routing is not taken into account. Similarly, the layout of digital control block comprising of an 8-bit thermometer decoder, is shown in fig.(5-12)(b). The digital logic is synthesized using 28nm standard cells. The layout is obtained after performing place-and-route of this netlist. The total digital area is $720 \mu m^2$. Adding both analog and digital area gives an estimate of the total layout area that would be occupied by the DAC. The total DAC area is around $2610 \mu m^2$.

Lastly, it is important to derive an estimate of the total power consumed by the DAC. There are 262 current sources each of 32nA running from 1.0V supply. Thus, the total analog power is 8.4 μW . The digital logic consumes negligible power as it works at a very slow update rate. Also, while estimating the total power, the contribution of bias circuit is ignored as it is shared with all the other DACs.

Table-5-4 summarizes important design specifications met by the 11-bit DAC design.

5-6 Summary

In this chapter critical sub-blocks for the timing error calibration loop are designed. The circuits are designed in industrial 28nm CMOS process and simulated using Cadence Spectre tool. Transistor level implementation of clock-path, track-and-hold, HOLD buffer and Digital-to-Analog converter (DAC) are discussed. First the implementation of clock-path was detailed. Its two key ingredients namely, clock phase generator and sampling edge tuning circuit are described. The clock phase generator generates all the critical signals for 2x time-interleaved ADC. Most importantly the masking clocks. These clocks are used to select the required edge from the main clock to further generate the respective lane's sampling clocks. This technique also enables shortest path between the main clock source and actual sampling switch. The sampling edge tuning circuit comprises mainly of the voltage-to-delay converting transistors and latching circuit that ensures the sampling switches are turned-off during the HOLD phase. Also, in order to keep the loading on clock-path for main-lane and reference lane balanced, additional dummy load is added in the clock path to compensate for 16x smaller bottom-plate sampling switch. Simulation of the clock-path show that its mean delay was about 11ps and it spreads by 0.5ps.

Second, the implementation of the track-and-hold for main-lane and reference lane is described. The designs achieve HD_3 of 72dB and 67.3dB for main-lane and reference lane respectively. The track bandwidth of 5.2GHz is achieved in simulations. However, the bandwidth for the reference lane does not exactly match with that of the main-lane and was off by 2.6MHz. The $3 \cdot \sigma$ spread in the timing error due to bandwidth mismatch between main-lane and the reference lane was 0.8ps.

Third, a HOLD buffer is designed to transfer voltage from a small (87.5fF) capacitor to a bigger capacitor. A fully-differential folded cascode OTA topology is chosen with switch capacitor CMFB. Simulations show that the OTA has a loop-gain of 61dB and phase margin of 78° . For 1GHz input signal the HD_3 of the HOLD buffer was around 40dB. Further, the Monte-Carlo simulations reveal that the buffer has an input offset of about 4.8mV. Both, input-offset and finite linearity of the HOLD buffer can have an adverse effect on the convergence of the loop. Hence, these non-idealities are modeled in MATLAB and simulations show that the performance of the calibration loop is unscathed in the presence of the HOLD buffer.

Lastly, the most important block namely, the DAC is designed. The expected dynamic range of the DAC was estimated to be around ± 3 ps. With the step-size of 5fs, this would require at least 10 bit resolution from the DAC. To implement the DAC, a fine-coarse architecture with 3 binary fine-bits and 8 thermometer coarse-bits is chosen for designing a 11-bit DAC (1bit for over-range). The functioning of the DAC is explained using a simple 3-bit implementation. Further, the implementation of unit cell together with its various design choices is also discussed. Circuit implementation details along with the biasing circuit and various design equations to compute vital DAC elements, are also presented. Simulations reveal that the DAC can achieve a LSB step-size of $240\mu\text{V}$ with a DNL of $\pm 0.6\text{LSB}$. The estimated area overhead and power penalty of the DAC are $2610\mu\text{m}^2$ and $8.4\mu\text{W}$, respectively.

Chapter 6

Conclusion

6-1 Problem Definition: A Recap

A universal radio transceiver that can be tuned to a carrier frequency over a wide range and can support any modulation over a wide range of data rates, would be extremely desirable to build tomorrow's mobile handsets. However, one of the major hurdles to overcome before this dream can be realized is the design of wideband capture ADC. Particularly, the specification of high sampling speed while supporting high resolution makes the designing of such wideband capture ADC extremely difficult.

In recent literature, time-interleaving architecture has shown great promise of achieving high sampling speed in an energy efficient way. In fact, they are already being applied to wideband capture ADCs for wire-line applications [36]. Unfortunately, implementation of this architecture comes with its own sets of issues. These interleaving issues are exacerbated for wireless systems due to the inevitable presence of adjacent channel interference or blocker tones. An image of these interfering tones is created due to interleaving issues (like timing error) which is later folded back into the desired signal band due sampling process of the ADC. Hence, minimizing these interleaving issues is mandatory for applying time-interleaving architecture for wideband capture ADCs.

Traditionally, the interleaving issues are minimized by employing calibration techniques. Along the same lines, the focus of this thesis has been on calibration of timing errors. As the in-band spurs due to timing errors should be below 90dB level, the target accuracy of timing error correction is 5fs or better. This thesis discusses techniques that can help achieve this level of accuracy.

6-2 Thesis contribution

Accurate detection of timing errors is extremely critical to achieve timing error correction accuracy of 5fs. However, timing errors do not lend themselves easily for detection as they are obscured by the input signal. In chapter-2 various topologies of timing error detection are studied. It was concluded that the most suitable architecture for high precision timing error detection is by using additional reference lanes. A recent publication employing this technique has shown encouraging results [11].

But still it is not clear if the correction accuracy of better than 5fs can be realized by using this architecture.

In this thesis the calibration loop proposed in [11] is studied. Some of the issues specific to the calibration loop are being identified. Relevant solutions are employed to overcome them. One of the critical finding was that the additional reference lanes load the input buffer. Also, this loading is not continuous and uniform but rather fluctuating as the reference lanes only occasionally sample the input signal. This fluctuation in the input buffer loading due to reference lane is a classical example of *Observer Effect*. For the time-interleaved ADCs this observer effect creates timing errors. In this thesis the observer effect of the reference lane is studied in a greater detail and relevant solutions are employed to solve it. The solutions presented in this thesis are either tested through MATLAB modeling or through circuit simulations.

More specifically, the key contributions of this thesis are as follows:

- **Making calibration loop DC error voltage resilient by embedding a high-pass filter.**

As shown in chapter-3, a DC error voltage is generated by various non-ideal mechanism like finite quantization of reference lanes, input-offset of reference lane, input-offset of HOLD buffer, and finite linearity of HOLD buffer. Depending on its amplitude, this error voltage can cause an error in steady state value of loop or can even make the loop non-convergent. This problem was solved by adding a high-pass filter to block this DC error voltage from being appearing at the input of the integrator. A MATLAB model was build to verify the effectiveness of this solution. Simulation results suggest that embedding a high-pass filter indeed helps to restore the steady-state value and ensure the loop convergence.

- **Identification of Observer effect due to loading of input buffer by reference lanes.**

The reference lane ADC loads the input buffer during the calibration phase. During the normal operation phase of the time-interleaved ADC they are not present. This absence and presence of the reference lanes to modulate the load of input buffer. In chapter-4 an equation was derived that estimates the timing error created due to this load modulation. It states that the timing errors are proportional to the output impedance of the buffer and the sampling capacitor. A mathematical estimation shows that these errors can be significant depending on the value of output impedance of the buffer and the reference lane sampling capacitor. In addition to load modulation, another observer effect of reference lane is the interaction between the sampling instance of the reference lane and the main lane. In chapter-4 it was established that this interaction is also a source of timing error.

- **Solving the sampling time interaction between main-lane and reference lane by adding delay lines.**

In chapter-4 the problem of sampling time interaction was brought out. This problem was solved by adding delay lines to the sampling front-ends of the main-lane and the reference lane. A model to estimate the delay of the delay lines made from metal wires is also given. It was shown that a wire of length $119\mu\text{m}$ long and $10\mu\text{m}$ wide can create an isolation of 2ps between the sampling instance of the main-lane and the reference lane.

- **Scaling down the reference lanes to solve the timing errors due to mismatch between dummy and reference lane.**

Additionally, in chapter-4 it was observed that though adding dummy lanes would solve the problem of buffer load modulation, the mismatch between the dummy lane and the reference

lane would give rise to additional timing errors. In order to suppress this additional source of timing errors, it was suggested that a scaled down version of the main-lane sampling front-end should be used for the reference lane and the dummy lane. A mathematical estimate is provided which can help designer to deduce the target spread in the reference lane and dummy lane sampling capacitor for a given specification on timing error. These mathematical expressions are cross-verified by appropriate simulation results. It was shown that to achieve a timing error correction accuracy of at least 5fs, the sampling front-end of dummy lane and reference lane both should be scaled down by at least 16x compared with the main-lane.

- **Design of sub-blocks to implement the calibration loop.**

The sub-blocks for the calibration loop were designed and implemented in chapter-5 using an industrial 28nm CMOS process. To achieve timing error correction within 5fs step-size, an 11-bit DAC is designed. The LSB of the DAC was about $240\mu V$. Monte-Carlo simulation suggest that the DNL of the DAC is $0.6 \cdot LSB$. As the reference lane was made 16x smaller, its the sampling capacitor cannot be used for charge redistribution and thus, quantization of the input signal. To surpass this limitation a HOLD buffer is employed which copies the voltage from a smaller sampling capacitor to a bigger one. The design of HOLD buffer operating at 50MHz is shown. Simulation results show that the HOLD buffer has an open-loop gain of 60dB, phase margin of 78° , and power consumption of 1.7mW. Further, a track-and-hold for main-lane and reference lane is designed which achieves HD_3 of 72dB for input frequencies of about 1GHz. The track-and-hold of the reference lane is 16x smaller and its bandwidth differs that of the main-lane only by 2.6MHz. Finally, in order to achieve smallest path between the clock source and actual sampling switch, a clock-path circuit is designed. The clock-path for the reference-lane would see 16x smaller load which can cause systematic timing error. To solve this issue the clock-path of the reference lane is loaded with 15 dummy switches.

6-3 Future Work

A summary of some potential future research is provided below:

- It would be very interesting to see the techniques discussed in this thesis actually implemented on a test chip. The real measurement data can address the question of how low the timing error spurs can be suppressed? Also it might even open up more interleaving related timing errors which are still out of grasp. Alternately, if the silicon works as expected and timing errors are indeed down to 5fs accuracy, then the problem of timing error which has been bugging interleaved ADCs since its inception, would be solved completely.
- If a silicon is implemented and due to some reason the timing errors are not corrected to a desired accuracy level, then additional technique of scrambling can also be deployed [36]. The scrambling can be implemented by randomly selecting one of the main-lanes rather than cycling through them. Also, the timing error due to mismatch between dummy lane and reference lane, can be reduced by scrambling or randomizing the dummy lanes.
- The stability and the convergence of the calibration loop was investigated primarily using a MATLAB simulation model. It would be highly desirable to have mathematically rigorous analysis by which the stability of calibration loop can be established.

- In a time-interleaved ADCs often multiple calibration loops operate simultaneous. For instance, offset, gain mismatch and timing error calibration loop all operate simultaneous. In such scenario, it is important to make these loops orthogonal especially the gain mismatch and the timing error loops. The gain mismatch should not appear as an error to timing error calibration loop and should be corrected only by gain calibration loop. In the calibration loop described in chapter-3, this orthogonality is readily available. If the error signal on which the gain calibration loop operates, is multiplied by $(1 - D)$ then the gain calibration and timing calibration loop can be made orthogonal. Few initial simulations show good promise but further exhaustive study needs to be performed.
- Finally, if higher speeds are required to be achieved then bandwidth mismatch of the sampling front-ends needs to be calibrated. This can be done by tuning either the common-mode voltage of the bottom-plate or the strapping potential of the bootstrapped top-plate switch.

Appendix A

Target Specifications

In this appendix target specifications for wideband capture ADC are estimated by taking an example of GSM cellular signal. As shown in fig.(A-1), a typical cellular signal is accompanied by a strong blocker tone. An image of this tone is generated due to interleaving issues of ADC like timing errors, which ends up in the signal band due to process of direct sampling at f_s . Also, ADC adds its own noise to the received signal which should be low enough to meet the GSM specifications.

To simultaneously capture all wireless standards existing in a mobile handset, the sampling rate of the ADC should be at least $5GHz$. Further, the receiver sensitivity for GSM (DCS-1800) cellular should be at least $-102dBm$ and in-band SNR required is $9dB$ (for $BER < 10^{-3}$). As an example both of these specifications are shown in fig.(A-1). Thus, the noise floor of the ADC over the Nyquist band ($0 - 2.5GHz$) should be below $-102dB - 9dB = 111dB$. Further, in wideband capture application the channels are separated in the digital domain by narrowing into one channel. For GSM cellular, the channel bandwidth, B_{CH} , is $200KHz$ which is narrowed from total capture spectrum of 0 to $2.5GHz$. The resulting oversampling gain, SNR_{OS} , is given by,

$$SNR_{OS} = 10 \cdot \log_{10}\left(\frac{f_s/2}{B_{CH}}\right) \approx 40dB \text{ where, } B_{CH} \text{ is the channel bandwidth} \quad (A-1)$$

Taking the over-sampling gain into account the in-band noise floor of the ADC should be at least $111dB - 40dB = 71dB$ i.e. 12 bits lower. Thus, the ENOB of the wideband capture ADC should be at least 12 bits.

Further, the linearity of the ADC is also critical. For GSM the input signal dynamic range is from $-102dBm$ to $-10dBm$. Assume a blocker signal at $-10dBm$ as shown in fig.(A-1). Due to interleaving issues (e.g. timing error) in a wideband capture ADC, an image of the blocker tone is created which folds back into the signal band after sampling. This image of blocker tone should be close to $-111dBm$ level i.e. should be below noise floor. Hence, the spurs created due to timing error in the interleaved-ADC should be below $-103dB$. As shown in Chapter-2, this translates timing error accuracy of less than $5fs$ RMS. Also, similar image of blocker tone will be generated due to distortion in the ADC. This argument implies that even the linearity of the ADC should be around $103dB$ level.

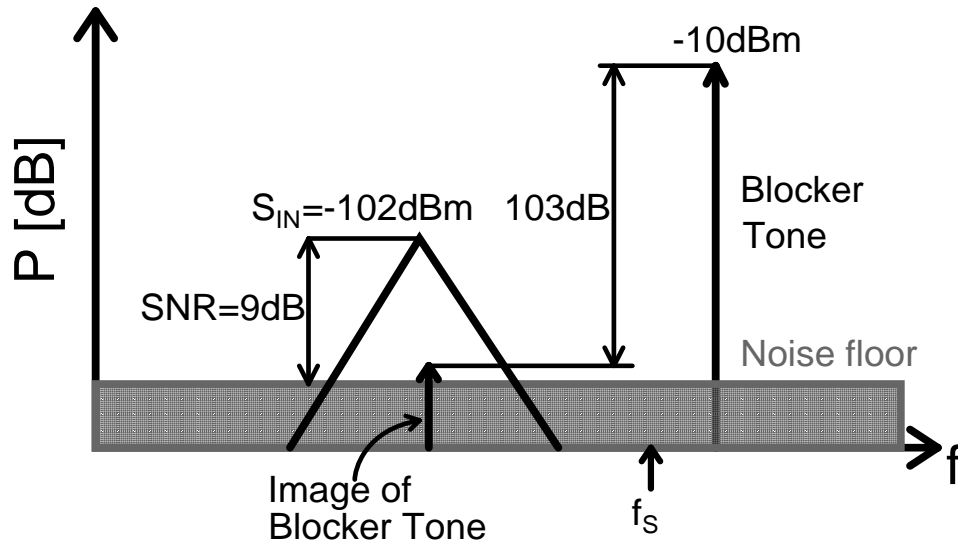


Figure A-1: Signal power level expected at the output of the wideband capture ADC.

Table A-1: Summary of specifications for wideband capture ADC for mobile applications

Parameter	Value	Remark
Sampling Frequency (f_s)	> 5 GHz	0-2.5GHz 1 st Nyquist-band
ENOB	> 12 bits	ADC in-band noise floor limit
Interleaving Spurs	< 103dB	Requires timing error < $5f_s$ RMS (Chapter-2)
Linearity	< 103dB	Similar requirement as on spurs due to interleaving
Power	20mW	As per literature [7]
FOM	1fJ/conv-step	22x better than state-of-the art design

Finally, the power budget on wideband ADC for mobile would be around 10mW-20mW [7]. For sampling speed of 5GHz and ENOB of about 12 bits, the FOM of the ADC should be around 1fJ/conv-step. Comparing this with 22.4fJ/conv-step achieved by state-of-the art ADCs in 65nm CMOS (fig.(1-2)(a)), it requires about 22x improvement.

Table-A-1 summarizes important specification for the wideband capture ADC which were argued above.

Appendix B

MATLAB Code to Simulate Basic Interleaving Issues

The following MATLAB code is used in Chapter-2 to study basic interleaving issues in time-interleaved ADCs.

```
% this code models ideal TI ADC by considering sampling instants
% in version-1 the sampling instants were static so we could not introduce
% timing errors
clear all;
clc;

Fs=5e9;
Ts=1/Fs;
NFFT=2^20;
Tmax = (NFFT-1)*Ts;
t=0:Ts:Tmax;

M = 104383; % adjust to change Fin values
Fin=M*(Fs/NFFT);

x1 = 0:4*Ts:Tmax;
x2 = Ts:4*Ts:Tmax;
x3 = 2*Ts:4*Ts:Tmax;
x4 = 3*Ts:4*Ts:Tmax;
Xin = 1*sin(2*pi*Fin*t);

%Enable following lines to disable error due to input offset error
VOS1 = 0;
VOS2 = 0;
```

```
VOS3 = 0;
VOS4 = 0;

%Enable following lines to enable error due to input offset error
% VOS1 = 0.3;
% VOS2 = 0.1;
% VOS3 = -0.15;
% VOS4 = -0.3;

%Enable following lines to disable gain mismatch
A1 = 1;
A2 = 1;
A3 = 1;
A4 = 1;

% Enable following lines to introduce gain mismatch
% A1 = 0.99;
% A2 = 1.015;
% A3 = 1.0;
% A4 = 1.009;

% Enable following line to disable timing mismatch
dTS = 0;
% Enable following line to enable timing mismatch
%dTS = 1.497e-12;

% Input sampling
y1 = A1*sin(2*pi*Fin*(x1+dTS))+VOS1;
y2 = A2*sin(2*pi*Fin*x2)+VOS2;
y3 = A3*sin(2*pi*Fin*x3)+VOS3;
y4 = A4*sin(2*pi*Fin*x4)+VOS4;

j = 1;
% output mux
for i=1:length(y1)
    VT(j) = y1(i);
    j = j+1;
    VT(j) = y2(i);
    j = j+1;
    VT(j) = y3(i);
    j = j+1;
    VT(j) = y4(i);
    j = j+1;
end

% Calculate Spectrum
X = fft(Xin,NFFT)/NFFT;
```



```
Y = fft(VT,NFFT)/NFFT;
SE = 1+NFFT/2;
f = (Fs/2e9)*linspace(0,1,SE);
Xdb = 20*log10(abs(X(1:SE)));
Ydb = 20*log10(abs(Y(1:SE)));
```

```
figure(1);
subplot(2,1,1);
plot(f,Xdb);
xlabel('Frequency [GHz]');
ylabel('Input Signal');
grid on;
subplot(2,1,2);
plot(f,Ydb);
xlabel('Frequency [GHz]');
ylabel('Output Signal');
grid on;
```

```
%Plot error signal
figure(2);
q = VT-Xin;
subplot(2,1,1);
plot(VT(1:50));
xlabel('Index');
ylabel('Output Signal');
grid on;
subplot(2,1,2);
plot(q(1:50));
xlabel('Index');
ylabel('Error Signal');
grid on;
```

Appendix C

Simulation Setup for Timing Errors

The method used to compute FFT and timing errors in Chapter-4 and Chapter-5 is detailed in this appendix. The expression given below are only applicable to Cadence SPECTRE tool.

C-1 Coherent FFT

The desired sampling frequency, f_S , is $50MHz$. Assume that number of FFT points (NFFT) is 128 and FFT window is rectangular. For this given information the frequency resolution of FFT, f_{BIN} is 390625. In order to achieve coherent FFT, the input frequency should be chosen such that it falls on the FFT grid i.e. an integer multiple of f_{BIN} . A non-integer number of periods of input signal would result in frequency leakage due to discontinuity at the edges of rectangular FFT window. Thus, f_{IN} should be,

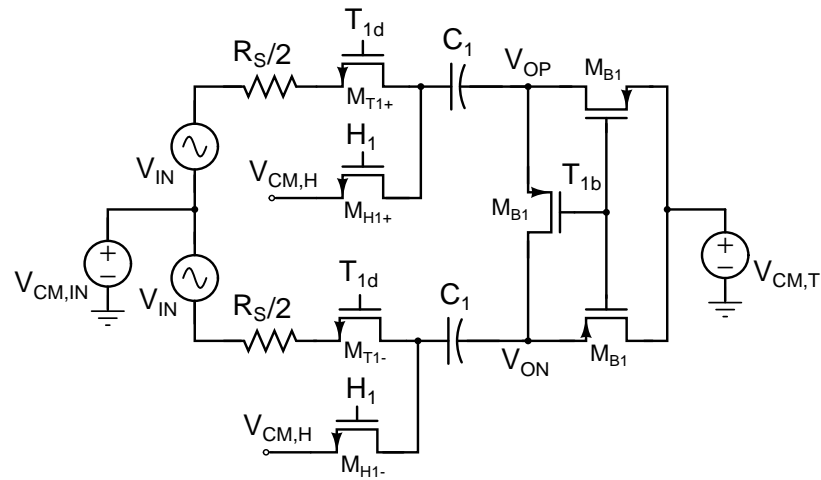
$$f_{IN} = M \cdot f_{BIN}, \text{ where } M = 2557 \quad (C-1)$$

$$= 998828125 (\approx 1GHz) \quad (C-2)$$

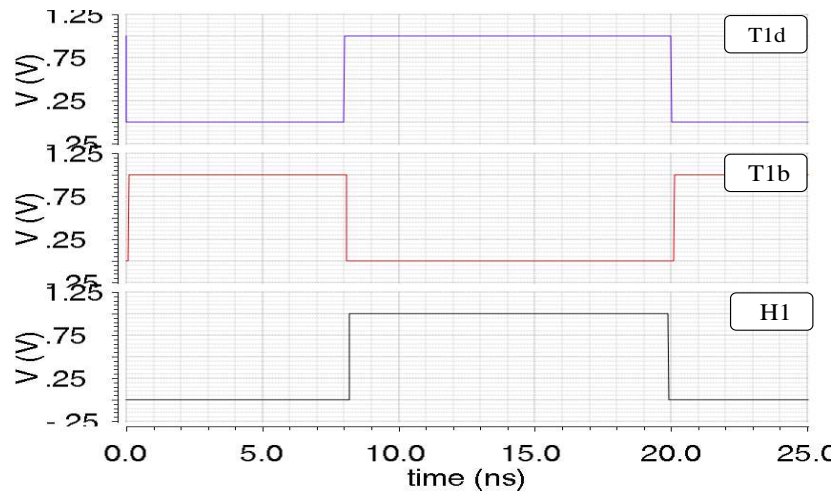
The value of integer multiplier, M is deliberately chosen to be a prime number because if M is even than the input signal sample pattern repeats itself and no additional information is obtained. Finally, $M = 2557$ is chosen as the desired input frequency should be close to $1GHz$.

C-2 Setting Up Variables

Fig.(C-1)(a) shows the fully-differential passive track-and-hold used in this thesis. It's design details can be found in Chapter-5. The input signal is sampled across sampling capacitor C_1 . The sampling instance is controlled by opening the bottom-plate switch M_{B1} by signal T_{1b} as shown in fig.(C-1)(b). To realize bottom-plate sampling, the top-plate switch, M_{T1+} opens after the bottom-plate switch. The differential output of the track-and-hold is defined by,



(a) Fully differential passive track-and-hold circuit.



(b) Timing Diagram.

Figure C-1: Track-and-hold circuit used in this thesis.

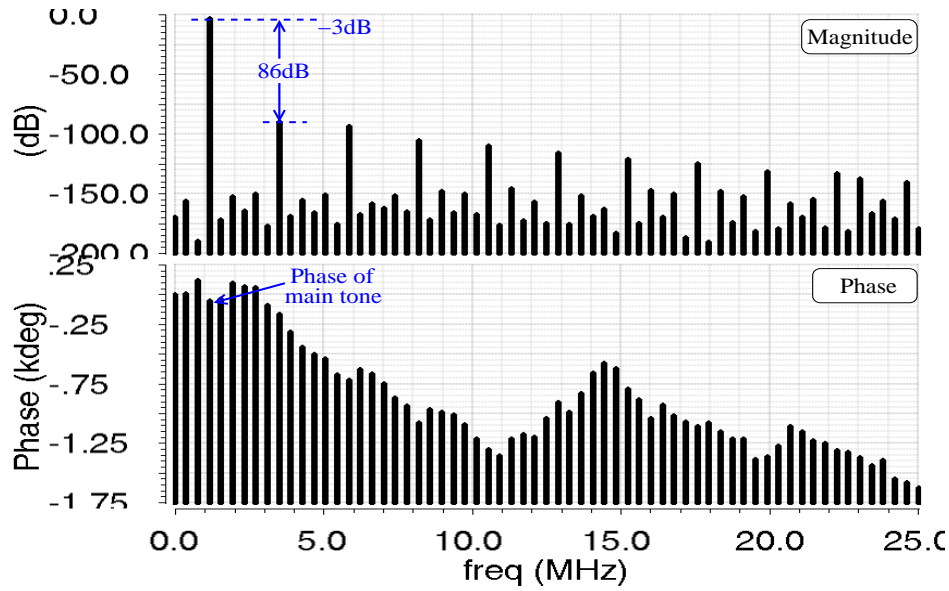


Figure C-2: Magnitude and phase plot for output of the track-and-hold circuit.

$$V_{OUT,DIFF} = (VT("/VOP") - VT("/VOP")) \quad (C-3)$$

This expression can be coded into SPECTRE. To estimate timing error, the phase of $V_{OUT,DIFF}$ should be known. In order to obtain phase information, FFT of $V_{OUT,DIFF}$ needs to be computed. As $V_{OUT,DIFF}$ is valid only in the HOLD phase, the FFT should also be computed in the HOLD phase. To compute the FFT assume that the first sample is picked at $7.75e - 08 \text{secs}$. For $N_{FFT} = 128$ points which are spaced with sampling period of $2e - 08$ (50MHz), the last sample should be located at $2.6375e - 06$ ($7.75 + 128 \cdot 2$). In order to ensure that this range of samples can be accommodated and sufficient FFT points are available, the first and last sample points of *sample* function should be defined appropriately. For example, as shown below,

$$V_{OUT,SAMP} = \text{sample}(V_{OUT,DIFF} \underbrace{1.885e - 08}_{\text{First Sample}} \underbrace{2.637e - 06}_{\text{Last Sample}}) \quad (C-4)$$

$$\text{"linear" } \underbrace{2e - 08}_{\text{Sampling period}}) \\ \text{PH1} = \text{phase}[\text{dft}(V_{OUT,SAMP} \underbrace{7.75e - 08}_{1^{st} \text{ FFT Sample}} \underbrace{2.6375e - 06}_{\text{Last FFT Sample}}) \underbrace{128}_{\text{No. of samples}} \text{"Rectangular"1 "Default")}] \quad (C-5)$$

Note that in order to ensure that samples are actually present at sampling points on which FFT is computed, the simulation variables like *maxstep* and *strobeperiod* must be set. For the simulation performed in this thesis both of these two variables are set to 5ps . Fig.(C-2) shows the simulated result of the two equations above for the track-and-hold circuit. To extract the timing information, first the phase of the main-tone needs to be obtained which can be done by using *value* function. Once

the phase information is obtained the timing error can be determined simply by dividing phase with the input frequency as given by,

$$\Delta\phi = \frac{\text{value(PH1 998828125)}}{360} \quad (\text{C-6})$$

$$\Delta t = \frac{\Delta\phi}{2\pi \cdot 998828125} \quad (\text{C-7})$$

Bibliography

- [1] J. Mitola, "The software radio architecture," *Communications Magazine, IEEE*, vol. 33, no. 5, pp. 26–38, 1995.
- [2] B. Murmann, "ADC Performance Survey 1997-2013," [Online]. Available: <http://www.stanford.edu/~murmman/adcsurvey.html>.
- [3] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara, and K. Kobayashi, "Explicit analysis of channel mismatch effects in time-interleaved ADC systems," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 48, no. 3, pp. 261–271, 2001.
- [4] R. Staszewski, R. Staszewski, T. Jung, T. Murphy, I. Bashir, O. Eliezer, K. Muhammad, and M. Entezari, "Software Assisted Digital RF Processor for Single-Chip GSM Radio in 90 nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 45, no. 2, pp. 276–288, 2010.
- [5] S. Mehta, D. Weber, M. Terrovitis, K. Onodera, M. Mack, B. Kaczynski, H. Samavati, S.-M. Jen, W. Si, M. Lee, K. Singh, S. Mendis, P. Husted, N. Zhang, B. McFarland, D. Su, T. Meng, and B. Wooley, "An 802.11g WLAN SoC," *Solid-State Circuits, IEEE Journal of*, vol. 40, no. 12, pp. 2483–2491, 2005.
- [6] W. Si, D. Weber, S. Abdollahi-Alibeik, M. Lee, R. Chang, H. Dogan, H. Gan, Y. Rajavi, S. Luschas, S. Ozgur, P. Husted, and M. Zargari, "A Single-Chip CMOS Bluetooth v2.1 Radio SoC," *Solid-State Circuits, IEEE Journal of*, vol. 43, no. 12, pp. 2896–2904, 2008.
- [7] R. Bagheri, A. Mirzaei, S. Chehrazi, M. Heidari, M. Lee, M. Mikhemar, W. Tang, and A. Abidi, "An 800-MHz ndash;6-GHz Software-Defined Wireless Receiver in 90-nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 12, pp. 2860–2876, 2006.
- [8] K. Doris, E. Janssen, C. Nani, A. Zanicopoulos, and G. Van Der Weide, "A 480 mW 2.6 GS/s 10b Time-Interleaved ADC With 48.5 dB SNDR up to Nyquist in 65 nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 46, no. 12, pp. 2821–2833, 2011.
- [9] J. Black, W.C. and D. Hodges, "Time interleaved converter arrays," *Solid-State Circuits, IEEE Journal of*, vol. 15, no. 6, pp. 1022–1029, 1980.

- [10] L. Sumanen, M. Waltari, and K. A. I. Halonen, "A 10-bit 200-MS/s CMOS parallel pipeline A/D converter," *Solid-State Circuits, IEEE Journal of*, vol. 36, no. 7, pp. 1048–1055, 2001.
- [11] D. Stepanovic and B. Nikolic, "A 2.8 GS/s 44.6 mW Time-Interleaved ADC Achieving 50.9 dB SNDR and 3 dB Effective Resolution Bandwidth of 1.5 GHz in 65 nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 48, no. 4, pp. 971–982, 2013.
- [12] B. Razavi, "Design Considerations for Interleaved ADCs," *Solid-State Circuits, IEEE Journal of*, vol. PP, no. 99, pp. 1–12, 2013.
- [13] Y. Oh and B. Murmann, "System embedded ADC calibration for OFDM receivers," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 53, no. 8, pp. 1693–1703, 2006.
- [14] J.-E. Eklund and F. Gustafsson, "Digital offset compensation of time-interleaved ADC using random chopper sampling," in *Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva. The 2000 IEEE International Symposium on*, vol. 3, 2000, pp. 447–450 vol.3.
- [15] J. McNeill, C. David, M. Coln, and R. Croughwell, "'Split ADC' Calibration for All-Digital Correction of Time-Interleaved ADC Errors," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 56, no. 5, pp. 344–348, 2009.
- [16] M. Mohsen and M. Dessouky, "13-bit 205 MS/s time-interleaved pipelined ADC with digital background calibration," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, 2010, pp. 1727–1730.
- [17] B. Peng, G. Huang, H. Li, P. Wan, and P. Lin, "A 48-mW, 12-bit, 150-MS/s pipelined ADC with digital calibration in 65nm CMOS," in *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, 2011, pp. 1–4.
- [18] P. Satarzadeh, B. Levy, and P. Hurst, "Bandwidth Mismatch Correction for a Two-Channel Time-Interleaved A/D Converter," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, 2007, pp. 1705–1708.
- [19] S. Gupta, M. Inerfield, and J. Wang, "A 1-GS/s 11-bit ADC With 55-dB SNDR, 250-mW Power Realized by a High Bandwidth Scalable Time-Interleaved Architecture," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 12, pp. 2650–2657, 2006.
- [20] K. Poulton, R. Neff, B. Setterberg, B. Wuppermann, T. Kopley, R. Jewett, J. Pernillo, C. Tan, and A. Montijo, "A 20 GS/s 8 b ADC with a 1 MB memory in 0.18 μm CMOS," in *Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International*, 2003, pp. 318–496 vol.1.
- [21] P. Schvan, J. Bach, C. Fait, P. Flemke, R. Gibbins, Y. Greshishchev, N. Ben-Hamida, D. Pollex, J. Sitch, S.-C. Wang, and J. Wolczanski, "A 24GS/s 6b ADC in 90nm CMOS," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, 2008, pp. 544–634.
- [22] Y. Greshishchev, J. Aguirre, M. Besson, R. Gibbins, C. Falt, P. Flemke, N. Ben-Hamida, D. Pollex, P. Schvan, and S.-C. Wang, "A 40GS/s 6b ADC in 65nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, 2010, pp. 390–391.

- [23] S. Jamal, D. Fu, N.-J. Chang, P. Hurst, and S. Lewis, "A 10-b 120-Msample/s time-interleaved analog-to-digital converter with digital background calibration," *Solid-State Circuits, IEEE Journal of*, vol. 37, no. 12, pp. 1618–1627, 2002.
- [24] S. Jamal, D. Fu, M. Singh, P. Hurst, and S. Lewis, "Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 51, no. 1, pp. 130–139, 2004.
- [25] H. Jin and E. Lee, "A digital-background calibration technique for minimizing timing-error effects in time-interleaved ADCs," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 47, no. 7, pp. 603–613, 2000.
- [26] A. Haftbaradaran and K. Martin, "A Sample-Time Error Compensation Technique for Time-Interleaved ADC Systems," in *Custom Integrated Circuits Conference, 2007. CICC '07. IEEE*, 2007, pp. 341–344.
- [27] S. Venkatesh, C. Anderson, R. Buehrer, and J. Reed, "On the use of Pilot-Assisted Matched Filtering in UWB Time-Interleaved Sampling," in *Ultra-Wideband, The 2006 IEEE 2006 International Conference on*, 2006, pp. 119–124.
- [28] M. El-Chammas and B. Murmann, "A 12-GS/s 81-mW 5-bit Time-Interleaved Flash ADC With Background Timing Skew Calibration," *Solid-State Circuits, IEEE Journal of*, vol. 46, no. 4, pp. 838–847, 2011.
- [29] D. Stepanovic, *Calibration Techniques for Time-Interleaved SAR A/D Converters*. PhD Thesis, University of California, Berkeley, 2012.
- [30] A. Muhtaroglu, G. Taylor, and T. Rahal-Arabi, "On-die droop detector for analog sensing of power supply noise," *Solid-State Circuits, IEEE Journal of*, vol. 39, no. 4, pp. 651–660, 2004.
- [31] J. Shor, K. Luria, and D. Zilberman, "Ratiometric BJT-based thermal sensor in 32nm and 22nm technologies," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, 2012, pp. 210–212.
- [32] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2008.
- [33] L. Schaper and D. Amey, "Improved Electrical Performance Required for Future MOS Packaging," *Components, Hybrids, and Manufacturing Technology, IEEE Transactions on*, vol. 6, no. 3, pp. 283–289, 1983.
- [34] M. Dessouky and A. Kaiser, "Very low-voltage digital-audio Delta; Sigma; modulator with 88-dB dynamic range using local switch bootstrapping," *Solid-State Circuits, IEEE Journal of*, vol. 36, no. 3, pp. 349–355, 2001.
- [35] C.-H. Lin and K. Bult, "A 10-b, 500-MSample/s CMOS DAC in 0.6 mm²," *Solid-State Circuits, IEEE Journal of*, vol. 33, no. 12, pp. 1948–1958, 1998.
- [36] E. Janssen, K. Doris, A. Zanakopoulos, A. Murrioni, G. van der Weide, Y. Lin, L. Alvado, F. Darthenay, and Y. Fregeais, "An 11b 3.6GS/s time-interleaved SAR ADC in 65nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, 2013, pp. 464–465.

