DIGITAL CALIBRATION TECHNIQUES OF PIPELINED AND ALGORITHMIC ADCS

Chinmaye Ramamurthy

Doctor of Philosophy Thesis

June 2022

International Institute of Information Technology, Bangalore

DIGITAL CALIBRATION TECHNIQUES OF PIPELINED AND ALGORITHMIC ADCS

Submitted to International Institute of Information Technology,
Bangalore
in Partial Fulfillment of
the Requirements for the Award of
Doctor of Philosophy

by

Chinmaye Ramamurthy
PH2017006

International Institute of Information Technology, Bangalore
June 2022

Dedicated to

family and friends

Thesis Certificate

This is to certify that the thesis titled **Digital Calibration Techniques of Pipelined and Algorithmic ADCs** submitted to the International Institute of Information Technology, Bangalore, for the award of the degree of **Doctor of Philosophy** is a bona fide record of the research work done by **Chinmaye Ramamurthy, PH2017006**, under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Subhajith Sen

Prof. Chetan Parikh

Bangalore,

The 20th of June, 2022.

DIGITAL CALIBRATION TECHNIQUES OF PIPELINED AND ALGORITHMIC ADCS

Abstract

Analog to Digital Converters (ADCs) act as a bridge between the real world analog signals and digital signal processors. ADCs convert continuous amplitude, continuous time signals to discrete amplitude, discrete time signals, which makes it easier for processing them. For the applications that require high speed and moderate resolution ADCs, pipelined ADCs, have become an obvious choice due to their many advantages like low complexity, low power dissipation and high conversion rate compared to other ADCs. With shrinking technologies, mismatches in the analog components used to build an ADC result in the digital output of the ADC having large distortion which require calibration. This thesis presents calibration techniques for improving the performance of the pipelined ADCs.

There have been many background and foreground calibration techniques proposed so far and they can be broadly classified into analog, statistical and deterministic techniques. Analog calibration techniques deal with tweaking the analog components to calibrate the ADCs and are more prone to error. Deterministic calibration techniques are suited for moderate resolution, moderate speed applications, while requiring minimal calibration hardware. Statistical Calibration techniques are suited for advanced technology node precision circuit designs where non-linearities cause significant errors. Calibration hardware generally consists of a microprocessor and many filters.

This thesis presents two background and four foreground digital calibration tech-

niques for pipelined ADCs. The proposed techniques have been tested on 1.5 bits/stage, 10/12 stages; 9/12 bits pipelined ADC designed using flip around MDAC where first 3-4 MSB stages are calibrated but can be used in higher resolution ADCs as well. The proposed calibration techniques can also be used to calibrate algorithmic ADCs with little or no modification in the calibration logic.

All the calibration techniques proposed in this thesis requires calculation of calibration coefficients which are used to obtain a calibrated digital output from the ADC. The first foreground calibration technique accounts for mismatches between the sampling and feedback capacitors and comparator offsets used in pipeline stages. Rest of the calibration techniques discussed in this thesis accounts for mismatches between the sampling and feedback capacitors, comparator offsets, finite op-amp gain, systematic offset errors and parasitic capacitance. The first and second calibration techniques neither requires a precise voltage generator nor probing the intermediate stages. The two calibration techniques can be used to calibrate any existing pipeline ADCs if there is an access to the digital output code of the pipeline stages. The third foreground calibration technique requires a reference ADC for calibration but very simple digital hardware. All calibration coefficients are accurate to 12 bits. The fourth calibration technique is based on an iterative algorithm which takes 5-10 cycles to calculate the calibration coefficients. The fifth and sixth calibration techniques are background deterministic calibration techniques which neither uses microprocessors nor does it take a longer time for calibration. Compared to other existing deterministic calibration techniques, the proposed techniques are characterized by its simplicity, minimal hardware and low power dissipation.

The circuits were implemented in Semiconductor Complex Limited, India (SCL)'s 180-nm CMOS technology, and were simulated in Cadence-AMS environment with the digital portions coded using Verilog HDL. In the proposed techniques, along with significant reduction in the INL and DNL in both pipelined and algorithmic ADCs after

calibration, there has been improvement in SNR and SFDR as well.

Acknowledgements

First of all, I would like to take this opportunity to extend my sincere gratitude to my Supervisor, Dr. Subhajit Sen and Co-Supervisor, Dr. Chetan Parikh for their patient guidance and continuous encouragement throughout my PhD work. It was all worth being their student and working under their guidance. Their thoughts have made a huge impact on my professional life and will always be a precious asset for my future career. I would also like to thank them for their constructive criticism which helped me a lot to improvise my research work.

This work was funded by the Indian Space Research Organization(ISRO) under RESPOND program. I would like to thank Dr. Hari Shankar Gupta and and Ms. Rinku Agarwal from ISRO for supporting this work.

I would like to express my gratitude and respect towards Prof Y. Gopala Rao for introducing me into this wonderful world of analog electronics and helping me realize that my potential is much more than what I had expected it to be.

I would like to thank Dr. S. Sadagopan, Former Director - IIITB and Dr. Debabrata Das, Director IIITB and Cmde SR Sridhar, Registrar, IIITB - Secretary to the Board, for all the financial support from the institution for the completion of my Ph.D.

I am in debt with much gratitude to the IT support team and Lab supervisors for providing good infrastructure, remote access even during the time of crisis (Covid-19) for the successful completion of my PhD with in time.

Not to forget the contributions of my PhD colleague Mr. P Surya. We have always worked together towards our goals from the beginning to the end motivating each other. I would like to thank him and all my IIITB friends for keeping the environment lively and joyful during my stay in IIITB.

I would like to thank all the wonderful people of Dept of ECE, RV College of Engineering, Dr Uma, Prof and Dean of student affairs, Dr Geetha, HOD of ECE Dept and Dr Uttara Kumari, Prof of ECE Dept for helping me obtain permission from RV College of Engineering to pursue my PhD at IIITB.

I express my gratitude towards Management of RVCE, Dr. K.N. Subramanya, Principal of RVCE for his cooperation towards permitting me to pursue my PhD at IIITB by providing me study leave for 3.5 years.

I extend my sincere thanks to all the Staff members of IIITB and RVCE who have encouraged me during my difficult times to complete my work successfully.

I am extremely indebted to my parents Mr. Rama Murthy N and Mrs. Vasundhara HK for their love, sacrifice and support. They are my first teachers who inspired me in all aspects of my life. I am extremely thankful for their wholehearted support and prayers. No words can explain the magnitude of support I received from my husband, Mr. Keshava Prasad, who has been a constant source of support and encouragement during the challenges of my doctral research life. I am truly thankful for having him in my life. My heartfelt thanks to my daughter, Nisarga Prasad, who has been extremely understanding and supportive during my PhD tenure.

Not to forget all my family and friends who have indirectly contributed towards the completion of my PhD by lifting up my spirits during my difficult times.

Last but not the least I would express my gratitude to God without which this project would not have been possible.

List of Publications

1. Chinmaye Ramamurthy, Chetan D Parikh and Subhajit Sen. ”Deterministic Digital Calibration Technique for 1.5 bits/stage Pipelined and Algorithmic ADCs with Finite Op-Amp gain and Large Capacitance Mismatches”. Circuits, Systems and signal processing, Springer, 40, pp 3684-3702, 2021. doi: 10.1007/s00034-021-01652-6

2. Chinmaye Ramamurthy, Surya P, Chetan D Parikh and Subhajit Sen. ”A Deterministic Digital Calibration Technique for Pipelined ADCs using a Non-nested Algorithm”. Analog IC design and Signal processing, Springer, 110, pp 557–568, 2022. doi:10.1007/s10470-021-01961-5

3. Chinmaye Ramamurthy, Chetan D Parikh and Subhajit Sen. ”Deterministic Digital Calibration of 1.5 bits/stage Pipelined ADCs by Direct Extraction of Calibration Coefficients”. 34th International Conference on VLSI Design, India, 2021. doi: 10.1109/VLSID51830.2021.00010

4. Chinmaye Ramamurthy, Subhajit Sen and Chetan D Parikh . ”Digital Calibration of 1.5 bits/stage Algorithmic ADC”. 18th International SoC Design Conference, South Korea, 2021. doi: 10.1109/ISOCC53507.2021.9614017

Contents

List of Figures

List of Tables

List of Abbreviations

ADC . . . . . Analog to Digital Converters

CMFB . . . . . Common Mode FeedBack

CMOS . . . . . Complementary Metal Oxide Semiconductor

ENOB . . . . . Effective Number of Bits

FFT . . . . . Fast Fourier Transform

DAC . . . . . Digital to Analog Converters

DNL . . . . . Differential Non Linearity

DR . . . . . Dynamic Range

GHz . . . . . Giga Hertz

INL . . . . . Integral Non Linearity

LSB . . . . . Least Significant Bit

MHz . . . . . Mega Hertz

MSB . . . . . Most Significant Bit

SFDR . . . . . Spurious Free Dynamic Range

SHA . . . . . Sample and Hold Amplifier

SNDR . . . . . Signal to Noise Distortion Ratio

CHAPTER 1

INTRODUCTION

1.1 Overview

All electronics system consists of sensors, digital signal processing unit and actuators. Sensors sense the incoming analog signals just like voice signals are sensed by a microphone, RF signals are sensed by antennas in the mobile phone etc. Due to low cost and low programmability, these signals are processed digitally and converted back to analog domain before they are output by the actuators. Therefore Analog to Digital Converter (ADC) plays a very important role in any electronics system. Among the various ADCs that are available today, pipelined ADCs are preferred due to its high speed and high conversion rate. State-of-the-art resolution specifications of pipelined ADCs range from 8 to 16 bits with sampling rates ranging from a few MHz to several GHz. Analog-front-ends (AFE) with this high bandwidth, high resolution specifications are required in a wide range of equipment such as communication, RADAR, instrumentation, sensing and imaging systems. For example, synthetic-aperture-radars and imaging cameras are useful in applications for defence, agriculture, urban planning, disaster management, etc. However, technology scaling has led to reduced supply voltage and decreased device gain which in turn has led to noise, distortion and non-linearity in the analog circuitry that limit the ADC performance. Also, small value capacitors for high speed and low chip area are required which increase capacitor mismatch. Hence

calibration of ADCs is very important to enhance the linearity of the ADCs. Several calibration techniques have been introduced in the past and they can be categorised into analog calibration techniques, statistical calibration techniques and deterministic calibration techniques. Analog calibration technique requires trimming of capacitors in the amplifiers to adjust the gain to the required value and it requires more number of cycles for calibrations. Hence, they are more prone to errors since they deal with tweaking the values of the analog components in the circuit. Statistical techniques are superior for calibrating ADCs in advanced technology nodes where the process-related errors are larger. But these techniques require a large amount of hardware and hence dissipate more power. They also require a larger number of clock cycles for calibration. Deterministic techniques measure the errors caused by non-ideal components, by comparing the actual digital outputs of the ADC with an ideal mathematical model, for known analog inputs. This error is then added to / subtracted from the digital output of the ADC during normal operation, thus correcting the error, and making the ADC output linear. Deterministic techniques require very little hardware and memory and are also faster in calibration, compared to statistical calibration techniques. Hence deterministic calibration techniques are generally preferred for moderate technology node designs. Calibration of ADC can be done either in background or in the foreground. Background calibration techniques do not disturb the actual working of the ADC since the calibration logic is run in parallel with the ADC where as foreground calibration techniques disturbs the actual working of the ADC during calibration.

The goal of this research is to develop novel digital calibration techniques to calibrate a pipelined ADC. Calibration algorithms have been demonstrated on 1.5 bits/stage, 10/12 stage pipelined ADC but these techniques can be used to calibrate high resolution ADCs as well. Six novel calibration techniques, out of which four foreground and two background calibration techniques have been implemented and are discussed here. All the techniques have been implemented in cadence in SCL 180nm technology.

- The first technique is a foreground calibration technique where the discontinuities due to jumps in the output transfer characteristics of the ADC (for a ramp input signal) caused by capacitor mismatches are calculated in all stages that needs calibration. This acts as calibration coefficients which are then added/ subtracted from the raw code of the ADC to get a calibrated digital output. This technique requires just a few adders and comparators for calibration. Considering 10 stages, 1.5 bits per stage pipeline ADC with upto 5% mismatches in the capacitances in the 3 most significant bit stages, INL of the pipelined ADC was reduced to less than 0.8 LSB after calibration; Signal-to-noise ratio improved by more than 10 dB and spurious-free dynamic range improved by 15 dB in the pipelined ADC. This calibration technique works well where the capacitor mismatches in the pipeline stage is 5% or less.

- The second technique is a foreground calibration technique where the calibration coefficients account for capacitor mismatch error which results in discontinuities and the finite op-amp gain error that results in the change of slope of the ADC output characteristics. The calibration coefficients are calculated and are later used to obtain calibrated digital output code of the ADC. This technique requires multipliers and dividers along with adders and comparators. Considering 10 stages, 1.5 bits per stage pipeline ADC with upto 10% mismatch in capacitances in the 3 MSB stages, the calibration technique improved SNR by more than 15 dB and SFDR by around 24 dB. INL of the pipelined ADC reduces to less than 0.7 LSB after calibration. This calibration technique works well for large capacitance mismatches. This technique has been slightly modified to calibrate one stage algorithmic ADCs.

- The third calibration technique is a foreground calibration technique. This technique directly extracts the weights with which the digital raw code of the pipelined ADC has to be multiplied, with minimal digital operations. The output digital

codes of the pipeline ADC are compared with that of the reference ADC to extract the weights directly. With calibration technique applied to the 12 stages pipeline ADC, it achieves SNDR of 62 dB @ input frequency equal to one third the sampling frequency and 71 dB at lower input frequencies and SFDR of 64 dB @ input frequency equal to one third the sampling frequency and 75 dB at lower input frequencies. Also, the measured INL and DNL of the calibrated ADC lie below 0.65 LSB and 0.3 LSB respectively.

- The fourth technique is foreground calibration technique implemented on 12 stages 1.5 bits/stage ADC. The calibration algorithm accounts for capacitor mismatch, offset errors, parasitic capacitances and finite open loop gain of the op-amp. A fixed point iterative algorithm is used to determine the weights, with which the digital codes of the pipeline stages of the ADC are multiplied, to obtain calibrated digital output. Unlike the state of the art calibration techniques, the proposed technique uses very less number of samples and cycles for calibration. The calibration algorithm has been implemented on 40 MSPS, 1.5 bits/stage; 12 bits pipeline ADC where 4 MSB stages are calibrated. For a 5% mismatch in capacitances in the 4 MSB stages of the pipelined ADC and open loop gain of the op-amp (A) equal to 52 dB, the calibration technique improved SNDR by more than 23 dB and SFDR by around 29 dB. The technique can also be slightly modified to calibrate algorithmic ADCs. For a 5% mismatch in capacitances of the algorithmic ADC and A = 52 dB, the proposed calibration technique improved SNDR by 22 dB and SFDR by around 29 dB.

- The fifth technique is background calibration technique implemented on 12 stages 1.5 bits/stage ADC. This requires an extra pipeline stage and an SHA. Each pipeline stage that requires calibration in the ADC is treated as one stage algorithmic ADC and calibration coefficients to correct capacitor mismatch and gain mismatch are calculated. These calibration coefficients are later used to get a

calibrated digital output of the ADC.

- The above algorithm can be slightly modified to calibrate 2 pipeline stages at a time. This technique doesn't require extra SHA but requires two pipeline stages. In both these techniques, during calibration, the pipeline stage that is being calibrated is replaced by the extra stage so that the actual working of the pipeline ADC is not affected. For a 6.25% mismatch in capacitances in the 4 MSB stages, both the calibration techniques improved SNR by more than 25 dB and SFDR by around 33 dB. INL of the pipelined ADC reduces to less than 0.67 LSB after calibration.

1.2 Thesis organization

Chapter 2 gives the literature survey of various calibration techniques that have been implemented so far. Chapter 3 to Chapter 7 discusses 6 novel calibration techniques of the pipelined ADCs and algorithmic ADCs along with the implementation of the calibration techniques and simulation results. Chapter 8 concludes the thesis.

CHAPTER 2

OVERVIEW OF CALIBRATION TECHNIQUES OF PIPELINED ADCS

The working of the ADC and various architectures of ADC are discussed in detail in [1]. Different architectures and sources of errors of pipeline ADCs are discussed in detail in [2]. In this chapter, we discuss various calibration techniques in literature used to calibrate pipelined ADCs.

Karanicolas [3], proposed a technique for 1 bit/stage ADCs which is very simple to implement, uses simple hardware and needs very little memory to store digital calibration co-efficients. The algorithm works as follows: To calibrate the n^{th} stage, first the analog input to that stage and the digital output of the stage are forced to 0. The residue output of the stage (say R1) is digitized by an ideal backend ADC. While keeping the analog input of the stage at 0, the digital output of the stage is forced to 1. From the digitized residue output (say R2), R1 is subtracted, and this gives the digital correction code of the n^{th} stage. Digital correction codes for all required stages are calculated similarly. These digital calibration codes are added to / subtracted from the raw ADC output to obtain a calibrated digital output. The limitations of this algorithm are: the gain of the first few (i.e. MSB) stages are intentionally reduced to less than 2 to avoid saturation of the subsequent stages in order to avoid missing decision levels. But this lowers the resolution of the ADC. The calculation of digital calibration co-efficients is

done in a nested manner starting from LSB to MSB of the stages to be calibrated. The technique accounts for capacitor mismatch error and not finite gain of the op-amp.

Sahoo and Razavi [4] described a calibration technique for a pipelined ADC that had a 4-bit resolution in the first stage, 1.5-bit resolution in the next 7 stages and 2 bits in the last stage. Digital calibration was carried out to account for capacitor mismatch error and finite op-amp gain error in the first stage and only capacitor mismatch errors in the next 4 stages. The technique requires generation of some precise analog voltages and hardware for digital multiplication and division.

Chuang and Sculley [5] describe a digital calibration technique for 1.5 bits/stage ADC similar to [3]. The stage gain of less than 2 reduces the resolution of ADC. A stage gain of greater than 2 saturates the output when the inputs are close to the reference voltage. This technique uses amplifiers with stage gain little greater than 2 which inturn reduces the input range. This technique uses simple hardware and very little memory for implementation.

Sahoo and Razavi [6] proposed a calibration technique that corrects the capacitor mismatch, finite op-amp gain error along with the error introduced due to the non linearity of the amplifier using LMS technique. This technique limits the input frequency to 94% of the Nyquist frequency. Also, this technique requires a lot of calculations to solve for 122 filter coefficients.

Most of the foreground calibration techniques makes use of calibration input of $\frac{V_{ref}}{4}$ for the calibration of each stage. C.Ravi [7] proposed a modification in the MX2 stage to obtain output residue voltage corresponding to an input of $\frac{V_{ref}}{4}$ that accounts for capacitor mismatches and finite op-amp gain of the particular stage with V_{ref} as the input voltage. This makes use of 2 extra capacitors in each MX2 stage instead of a standalone resistor ladder to provide the calibration inputs. This technique still requires external controlling and interruption of the ADC during the calibration.

V.Sarma [8] uses dithering in the MX2 stages to reduce the quantization noise in the ADC. Although the SFDR of the ADC is close to the theoretical limits, this requires extra hardware like the LFSR for pseudo random noise generation.

Yun-Shiang Shu [9] uses dithering technique to measure the capacitor mismatch errors and op-amp gain errors in the background. Dither added to the signal during calibration is signal dependent so that the actual signal range is not affected due to the addition of the dither. This again requires extra hardware and a lot of testing so that the signal does not go out of the range.

Carl R. Grace [10] proposed an adaptive calibration technique for pipelined ADCs that accounts for both gain mismatch and non-linearity of the ADC. This technique requires an embedded microprocessor for calibration and takes a long time to converge. Also this calibration technique holds good for MX2 stages built from non flip around MDAC.

X. Wang [11] and Y. Chiu [12] proposed background adaptive calibration technique where the output of the ADC is compared with a slow and accurate ADC and calibration coefficients are calculated based on the error output. This technique requires calibration of the reference ADC.

Nan Sun [13] proposed a calibration technique to correct the errors due capacitor mismatch by swapping the capacitors depending of the digital output bit. But, this technique works well only for ADCs with less than 4% capacitor mismatch. This also requires extra capacitors for calibration.

Ozan E. Erdogan [14] proposed a simple adaptive algorithm to calibrate 1 bit/stage algorithmic ADC. This technique uses very little hardware but takes a longer time to converge. Also, this calibration algorithm can be applied only to ADCs with non flip around DAC.

Jipeng Li [15] estimates the calibration co-efficient in each stage with the help of PN sequence by extracting the radix from each stage using digital correlation method.

H. Adel [16], proposed a split ADC calibration scheme where the slope of one ADC is used in the calibration of the other. Although this does not involve any feedback or iterations, this requires 2 similar ADCs and hence larger area.

The next 5 chapters proposes novel digital calibration techniques for pipeline and algorithmic ADCs and also showcases the simulation results.

CHAPTER 3

A DETERMINISTIC DIGITAL CALIBRATION TECHNIQUE FOR PIPELINED ADCS USING A NON-NESTED ALGORITHM

This chapter proposes a novel deterministic technique to digitally calibrate 1.5-bits/stage and 1-bit/stage pipelined as well as algorithmic ADCs. The technique accounts for mismatches between the sampling and feedback capacitors, comparator offsets and charge injection. Compared to other existing deterministic calibration techniques, this technique is characterized by its simplicity, minimal hardware and low power dissipation. The technique is demonstrated on a 1.5-bits/stage, 10-stage pipelined ADC, in a 180-nm CMOS technology. The pipeline hardware is neither interrupted nor externally controlled during calibration; the technique can handle both missing codes and missing decision levels (i.e. MDAC gains less than 2 as well as greater than 2). For 5% mismatches in the capacitances in the 3 most significant bit stages, there was a significant reduction in the integral nonlinearity in both pipelined and algorithmic ADCs after calibration; Signal-to-noise ratio improved by 10dB and spurious-free dynamic range improved by 15dB in the pipelined ADC.

3.1 Proposed Calibration Technique

To illustrate the technique, a 1.5-bits per stage, 10-stages fully-differential pipelined ADC is considered, of which the first 3 stages are calibrated, and stages 4 to 10 are assumed to be ideal [3]. The input analog voltage is assumed to have the range –1.2 V to 1.2 V. Nominally, the MDAC gain of every stage is designed to be 2. But due to capacitance mismatch, the gain of a stage may be lower or higher. The former leads to missing codes, and the latter leads to missing decision levels [5]. Capacitance mismatches reported in [7] is 3%. The SCL foundry has capacitance mismatches in the range of 2-3%. Advanced technology node helps us to build capacitor with smaller area which in turn leads to larger mismatch [17]. Unfortunately, deterministic calibration techniques, such as the one proposed in this chapter, cannot correct for missing decision levels (gains > 2 in 1 bit/stage pipelined ADCs) [3]; they can correct only for missing codes (gains < 2). Therefore the technique proposed here performs calibration in two steps. In the first step, it is determined if the gain of each stage to be calibrated is less than or greater than 2. If it is greater than 2, then the circuit of the MDAC is modified so as to make the gain less than 2. To first determine the gains, a finely incremented ramp voltage is input to the ADC from 0 to $\frac{V_{ref}}{4}$. As the input voltage is ramped up, when the N^{th} output code changes from 0 to +1, an increased gain of the N^{th} stage MDAC causes a negative jump in the output characteristics and a decreased gain causes a positive jump [5]. This is illustrated in Figure FC3.1, for a gain error in the first stage (due to capacitance mismatch) and an ideal backend ADC. If the gain of a particular MDAC stage is greater than 2, then the capacitors in the MDAC (also called as MX2) of that stage are swapped, so that the stage gain becomes less than 2. This is done as shown in Figure FC3.2, which shows a modified MX2 circuit that enables the swapping of capacitors.

A conventional MX2 circuit (for a gain of 2) will not have the switches marked with

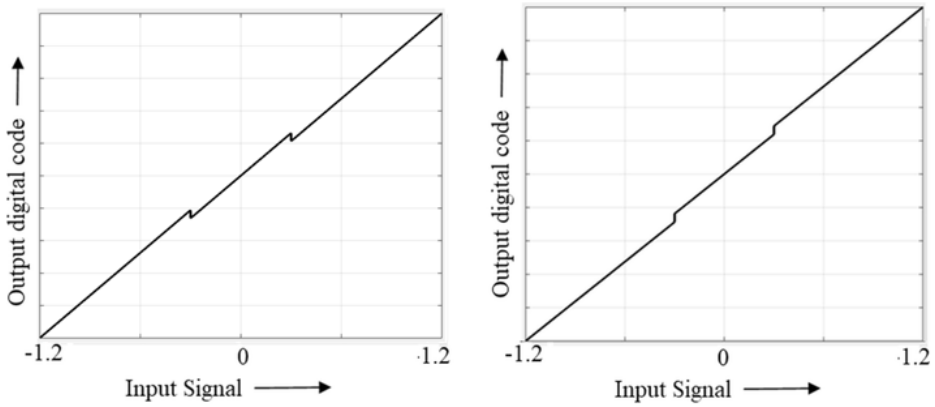

Figure FC3.1: 1.5 bits/stage pipeline ADC output characteristics with first stage gain (a) greater than 2, and (b) less than 2

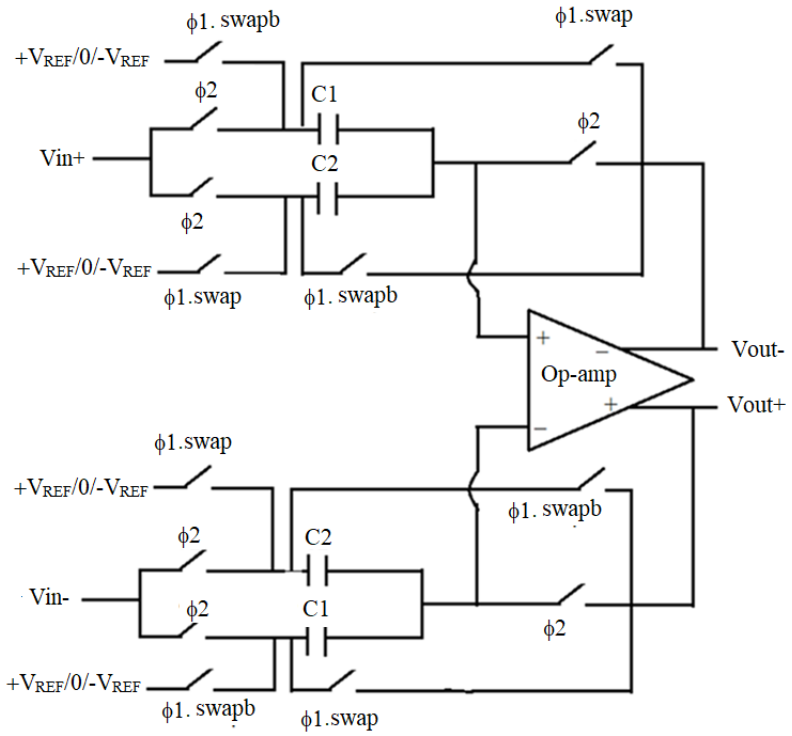

Figure FC3.2: MX2 circuit of the Pipelined ADC of the stages that require calibration

a ‘swapb’ clock signal, and its gain will be (1+$\frac{C2}{C1}$), with C1 = C2 in the ideal case. If due to capacitance mismatch, C2 $>$ C1, then this gain will be greater than 2. In such a case, in the circuit of Figure FC3.2, the clock signals ‘ϕ1swapb’ are made high, rather than the signals ‘ϕ1swap’. This causes the gain to be (1+$\frac{C1}{C2}$), which becomes less than 2 [5]. The residue output, V_{out}, with signal *swapb* high is given by Eqn 3.1.

$$V_{res,n} = \frac{(C_1(n)+C_2(n)).V_{in}(n) - V_{ref}.C_1(n).D_n}{C_2(n)} \qquad \text{(Eqn 3.1)}$$

The residue output, V_{out}, with signal *swap* high is given by Eqn 3.2.

$$V_{res,n} = \frac{(C_1(n)+C_2(n)).V_{in}(n) - V_{ref}.C_2(n).D_n}{C_1(n)} \qquad \text{(Eqn 3.2)}$$

Once this is done for all the stages that are to be calibrated, each of them will have a gain less than 2, and the digital calibration technique described below can then be applied. This process of swapping the capacitors eliminates the need for intentionally forcing the gain of each stage to be less than 2 by design, as was done in [3]. The latter caused a reduction in the resolution of the ADC, which the present technique does not suffer from. Note that only the stages that are to be calibrated need to have the modified MX2 circuit of Figure FC3.2. The rest of the stages will have the conventional MX2 circuit.

We now discuss the second step of the proposed calibration algorithm. For the 1.5-bit per stage architecture being considered, the output voltage (V_{out}) of the MX2 stage (assuming that the op-amp open-loop gain is large) is given by Eqn 3.3 [5]

$$V_{out} = (1+\beta).V_{in} - B.V_{ref}.\beta \qquad \text{(Eqn 3.3)}$$

Where B = 1 if Vin $>$ $\frac{V_{ref}}{4}$, B = –1 if Vin $<$ -$\frac{V_{ref}}{4}$ and B = 0 otherwise, V_{ref} is the reference voltage and β is the inverse of the signal gain.

Let $\beta 1$, $\beta 2$ and $\beta 3$ be the capacitance ratios of stage1, stage2 and stage3, respectively. With each of these three stages having a gain less than or equal to 2, a capacitance mismatch will cause an upward jump in the transfer characteristics. The goal now is to determine the height of each jump, and then correct it in the digital output of the ADC. We begin by noting that a gain error in the n^{th} stage causes a jump in the output characteristics when the n^{th} output digital code changes from –1 to 0 or from 0 to +1. This is shown in Figure FC3.3 for capacitance mismatches in the first three MSB stages, where a capacitance mismatch of 10% is considered for each of these stages. Note that as the digital output traverses from all –1's to all +1's, the MSB stage causes 2 jumps in the transfer characteristics (indicated by triangles in Figure FC3.3), the second MSB causes 6 jumps (shown by rectangles in Figure FC3.3), and the third MSB causes 14 jumps (shown by circles in Figure FC3.3), so that there are a total 23 segments in the characteristics. The jumps caused by a particular stage have the same height, because

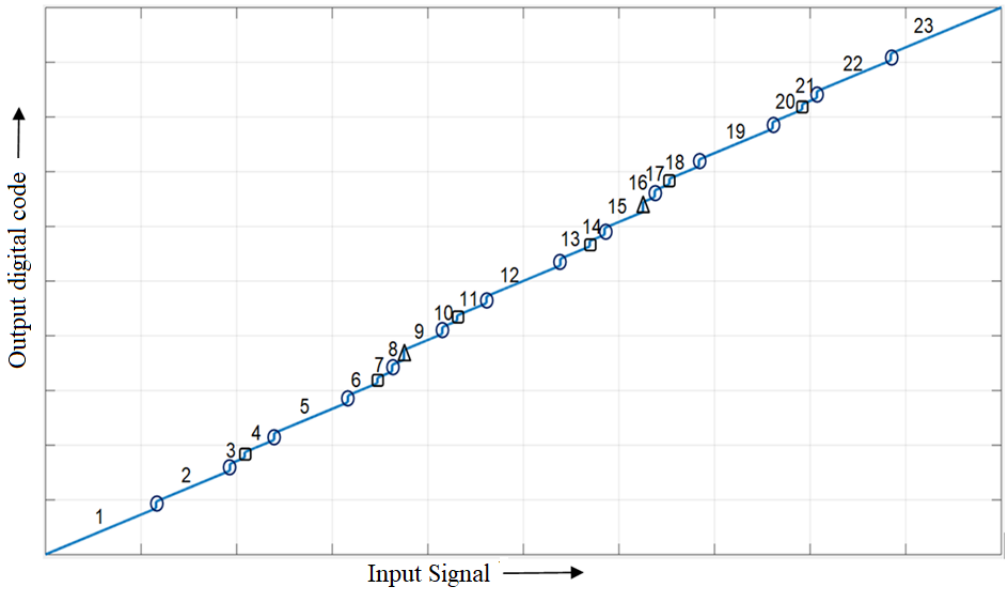

Figure FC3.3: Output characteristics of ADC for the capacitor mismatch in 3 MSB stages

this height is a function of only the capacitance mismatch of that stage. Thus only one jump per stage needs to be measured. Let S1, S2 and S3 be the heights of the jumps due to stages 1, 2 and 3 respectively. We calculate S3 for an analog input of around

75mV ($\approx \frac{V_{ref}}{16}$), which corresponds to an input to the 3^{rd} stage of 300mV ($=\frac{V_{ref}}{4}$). At this voltage, the 3^{rd} MSB changes from 0 to +1. The analog equivalent ($V_{3,1}$) of the ADC digital output when the input to stage3 is $\frac{V_{ref}}{4}$ and the 3^{rd} MSB is +1 is given by [3] Eqn 3.4.

$$V_{3,1} = \frac{V_{ref}}{8} + \frac{(1+\beta 3).\frac{V_{ref}}{4} - \beta 3.V_{ref}}{8} \qquad \text{(Eqn 3.4)}$$

Similarly, the analog equivalent ($V_{3,0}$) of the ADC digital output when the input to stage3 is $\frac{V_{ref}}{4}$ and the 3^{rd} MSB is 0 is given by Eqn 3.5.

$$V_{3,0} = \frac{(1+\beta 3).\frac{V_{ref}}{4}}{8} \qquad \text{(Eqn 3.5)}$$

Thus the analog equivalent of the jump in the output when the 3^{rd} MSB transitions from +1 to 0, is given by Eqn 3.6.

$$S3, analog = V_{3,1} - V_{3,0} = \frac{V_{ref}}{8} - \frac{\beta 3.V_{ref}}{8} \qquad \text{(Eqn 3.6)}$$

Note that ideally, $\beta 3$ = 1, and then S3, analog = 0V, which is as expected. Similarly, the jump due to the 2^{nd} MSB is determined at analog input of around 150 mV, and can be written as S2,analog = $\frac{V_{ref}}{4}$ - $\frac{\beta 2.V_{ref}}{4}$, and the jump due to the MSB is determined at 300 mV, and can be written as, S1,analog = $\frac{V_{ref}}{2}$ - $\frac{\beta 1.V_{ref}}{2}$. During calibration, the hardware required for digital computations consist of a few comparators and adder / subtractor. Hardware complexity is highly reduced since no division and multiplication operations are necessary. Calibration mode ends when the calibration co-efficients for all the stages to be calibrated are calculated.

Having determined the jumps (S1, S2 and S3), we now use them to perform correction in the digital output of the ADC (cf. Figure FC3.3). The goal of the correction algorithm is to obtain a smooth straight line [3], [5]. To achieve this, any of the 23

segments in Figure FC3.3 is taken as a reference segment, and for all other segments an appropriate linear combination of S1, S2 and S3 is added/subtracted to eliminate the jumps. We choose segment 12 as the reference, as this requires the minimum hardware for error correction. Starting with segment 12, segments 11 and 13 encounter jumps due to the capacitance mismatch of the 3^{rd} MSB stage. So S3 is added to the digital codes belonging to segment 11 and subtracted from the digital codes of segment 13, to align them with segment 12. Similarly (S3 + S2) is added to the digital codes of segment 10 and subtracted from the digital codes of segment 14, and so on. The complete set of corrections is shown in Table TC3.1. With these corrections, all the jumps due to capacitor mismatches are eliminated and the output characteristic of the ADC is made linear.

A bird's eye view of the calibration algorithm for the capacitance mismatch in the first stage is shown in Figure FC3.4 [18].

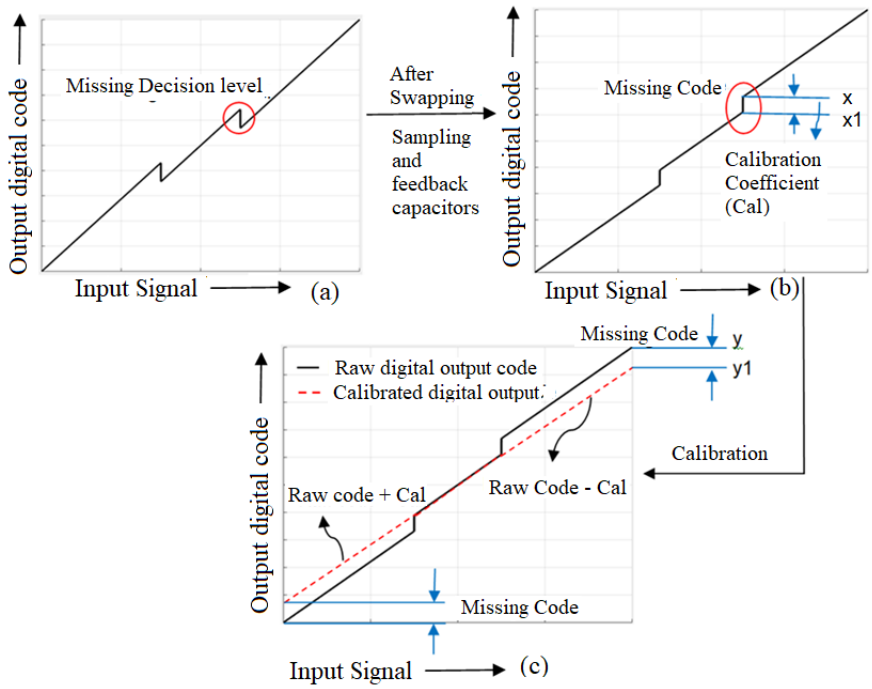

Figure FC3.4: Bird's eye view of the calibration algorithm to calibrate the first stage

Table TC3.1: Correction magnitudes for each of the segments of FC3.3

Segment	D1	D2	D3	Correction
1	00	00	00	S1*1+S2*3+S3*7
2	00	00	01	S1*1+S2*3+S3*6
3	00	00	10	S1*1+S2*3+S3*5
4	00	01	00	S1*1+S2*2+S3*5
5	00	01	01	S1*1+S2*2+S3*4
6	00	01	10	S1*1+S2*2+S3*3
7	00	10	00	S1*1+S2*1+S3*3
8	00	10	01	S1*1+S2*1+S3*2
9	01	00	01	S2*1+S3*2
10	01	00	10	S2*1+S3*1
11	01	01	00	S3*1
12	01	01	01	0
13	01	01	10	-(S3*1)
14	01	10	00	-(S2*1+S3*1)
15	01	10	01	-(S2*1+S3*2)
16	10	00	01	-(S1*1+S2*1+S3*2)
17	10	00	10	-(S1*1+S2*1+S3*3)
18	10	01	00	-(S1*1+S2*2+S3*3)
19	10	01	01	-(S1*1+S2*2+S3*4)
20	10	01	10	-(S1*1+S2*2+S3*5)
21	10	10	00	-(S1*1+S2*3+S3*5)
22	10	10	01	-(S1*1+S2*3+S3*6)
23	10	10	10	-(S1*1+S2*3+S3*7)

Figure FC3.4(a) shows the output characteristics of the ADC for a first stage gain greater than 2. Figure FC3.4(b) shows the output characteristics for a first stage gain less than 2, after the sampling capacitor and feedback capacitors are swapped. The calibration coefficients are calculated by subtracting x1 from x as shown in Figure FC3.4(b). Figure FC3.4(c) shows a graph of the calibrated output digital code of the ADC after adding/subtracting the calibration coefficients from the raw code of the ADC. The calibration algorithm improves the linearity of the ADC, but it results in missing codes (y-y1) near the rails [18].

The proposed algorithm can also be used to calibrate 1 bit/stage pipelined ADCs, as well as algorithmic ADCs. For algorithmic ADCs, which is a single 1-bit pipelined stage used in a cyclic fashion, there is only one stage, and only one MX2 circuit, and

therefore only a single-step calibration is necessary. The procedure for calibration is as follows: Force the input voltage to 0 V and the first digital output bit to 1. Then, let the algorithmic ADC calculate the rest of the bits. At the end of (N-1) cycles, the (N-1) bits are output. Let the N-bit output (with MSB=1) be called R1. Find the complement of R1 (say R1'). Then (R1 – R1') is the digital calibration code equivalent to S1 of the pipelined ADC. Similarly, let R2 be the first, i.e. most significant, (N-1) bits of R1, and let R2' be the complement of R2. Then (R2 – R2') gives the digital calibration code equivalent to S2 of the pipelined ADC. This process is repeated for as many bits as is desired. The higher the number of bits calibrated, the better is the INL, but the lower is the resolution [3]. The flowchart to calculate the calibration coefficients of the 'N' MSB stages to be calibrated in a pipeline ADC is as shown in Figure FC3.5 [18].

During normal operation of the ADC, correction magnitudes are calculated as discussed earlier in this section, and these are added / subtracted from the raw code to get the calibrated digital output. The algorithm can also be applied to calibrate a 1.5 bits/stage algorithmic ADC with the input voltage forced to 0.3V and first digital output bit forced to +1.

3.2 Implementation and Simulation Results

The circuit was implemented in Semiconductor Complex Limited, India (SCL)'s 180-nm CMOS technology, and was simulated in Cadence-Spectre. As mentioned in section.3.1, a 1.5 bit/stage, 10-stage pipelined ADC was implemented as a test case. The 1.5-bit flash ADC in each stage was designed with Strong-ARM latch comparators [19]. The reference voltages ($\pm\frac{V_{ref}}{4}$) for the comparator were generated using switched-capacitor voltage divider circuit [20]. The circuit of the multiply-by-2 DAC (MX2) was discussed in 3.1 (see Figure FC3.2). The op-amp in the MX2 was a two-stage op-amp with a telescopic cascode first-stage followed by a push pull common source stage, and

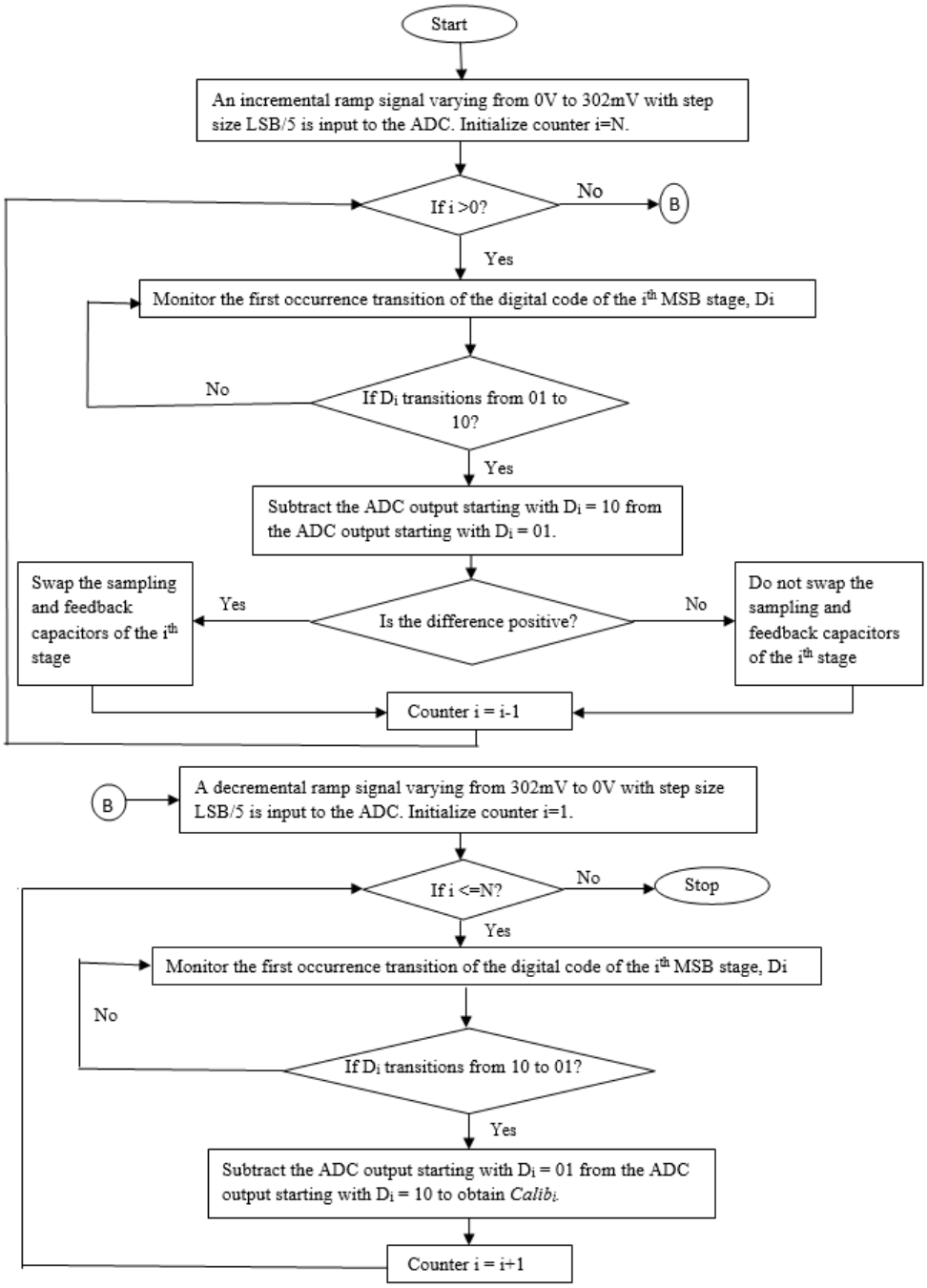

Figure FC3.5: Flowchart to calculate the calibration coefficients

common-mode feedback for each stage. The op-amp was designed to provide a gain of 87dB, a unity-gain frequency of 500MHz, and a phase margin of 56^o over all process corners [Appendix A.1].

The hardware required for calibration consisted of the digital block needed to calculate the correction codes of TC6.3, and a circuit to generate a ramp voltage. The block level representation of the calibration algorithm that calculates the calibration coefficients of the first MX2 stage is shown in Figure FC3.6. This circuit was used for all the stages that were calibrated. In Figure FC3.6, the input *D [9:0]* represents the 10 bits

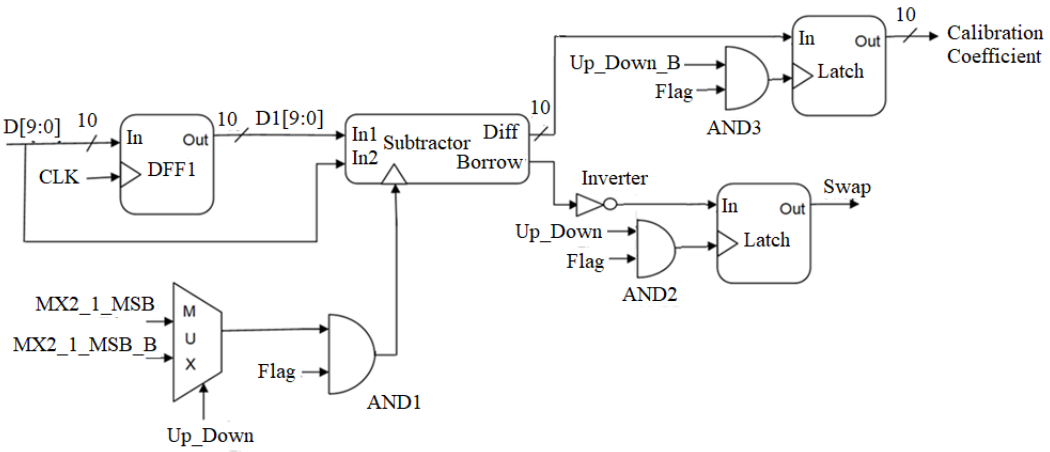

Figure FC3.6: Block diagram of the calibration algorithm

raw digital output from the ADC before calibration. It is known that the digital output code from each MX2 stage can take a value of +1, 0 or -1 (cf. Eqn 3.3) which is digitally represented using 2 bits. The MX2 digital output code '+1' is represented by the digital bits '10', '0' is '01' and '-1' by '00'. The input bit $MX2\_1\_MSB$ represents the MSB of the first MX2 stage which takes a value of '0' if the digital output code of the MX2 stage is '0' or '-1', and takes a value of '1' if the digital output code of the MX2 stage is '+1'. The bit $MX2\_1\_MSB\_B$ is a complement of $MX2\_1\_MSB$. The signal Up_Down, if set to '1', indicates that the input signal is ramping up (to swap the sampling and feedback capacitors if the stage gain is greater than 2), else the signal is set to 0 when the input signal ramps down (to calculate the calibration co-efficients). The

output signals *Swap* and *Calibration Coefficient* are latched once during the calibration cycle. The former is latched when the input $MX2\_1\_MSB$ changes from 0 to 1 during input signal ramp-up and the latter is latched when the bit $MX2\_1\_MSB$ changes from 1 to 0 during ramp-down. Any further modification to the output signals is prevented by setting the $flag$ bit to low (initially set to high) after the output signals are latched once. During the calibration phase, with input signal ramp-up (Up_Down = 1, flag = 1), when $MX2\_1\_MSB$ bit changes from 0 to 1, the corresponding ADC output *D [9:0]*, is subtracted from the delayed ADC output *D1 [9:0]*. If the subtraction operation sets the *Borrow* bit low, indicating a stage gain greater than 2, then, *swap* signal is set high, else *swap* signal is set low indicating that the stage gain is less than 2. At this point, the *flag* bit is set low so that the *Swap* signal remains unchanged after it is latched once. With decremented ramp (Up_Down = 0, flag = 1), when $MX2\_1\_MSB$ bit changes from 1 to 0, the corresponding ADC output *D [9:0]* is subtracted from the delayed ADC output *D1 [9:0]* to calculate the calibration coefficients. At this point, the *flag* bit is set low so that the calibration coefficients remain unchanged after it is latched once. This marks the end of calibration phase.

The ramp voltage that is required for calibration needs to have a step size of $\frac{LSB}{5}$ and was generated using a switched capacitor integrator [21]. As this calibration technique is a foreground technique, the same (high-performance) op-amp was used for the sample-and-hold circuit during normal operation and for the switched-capacitor integrator during calibration. The circuit of the combined block is shown in Figure FC3.7. C_H, ϕ1.calb and ϕ2.calb form the sample-and-hold circuit, and C_C, C_F, C_{FE}, ϕ1.cal and ϕ2.cal form the calibration circuit. ϕ1 and ϕ2 are non-overlapping clocks. Switches S1 and S2 are connected to *calp* and *caln*, respectively, during calibration and to *calpb* and *calnb*, respectively, during normal operation. Recalling from section 3.2 that the ramp is used to determine the three jumps in the transfer characteristics due to the mismatches in the three MSB stages, the ramp needs to be extremely fine in the vicinity

of the three jumps of interest, and can be coarse while traversing from one jump to the next. A fine ramp throughout will also work, but will take an inordinately long time to process. Thus in the switched capacitor integrator, C_F provides a fine ramp, and C_C is added in parallel to obtain a coarse ramp, with C_F set approximately to $\frac{C_C}{100}$.

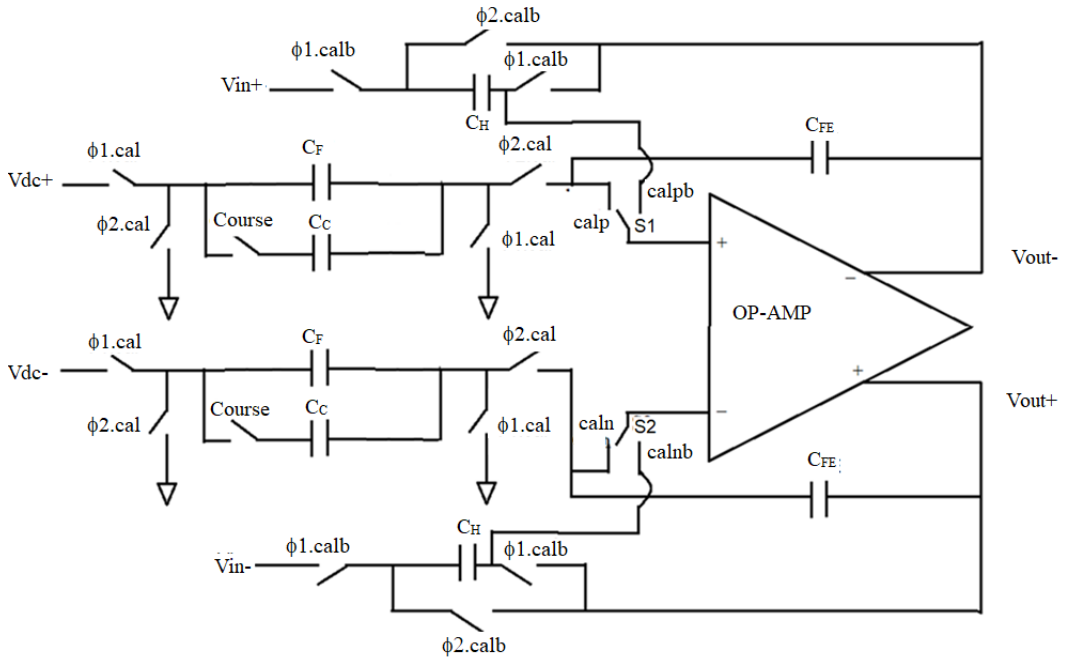

Figure FC3.7: Sample and hold amplifier and switched capacitor integrator

The ramp signal needs to traverse the range from 0V to 302mV ($\approx \frac{V_{ref}}{4}$) for calibration. The range of the fine ramp that is to be generated varies depending on the maximum gain error considered. For this design, fine ramp is required to be generated between 75mV$\pm$15mV, 150mV$\pm$10mV and 300mV$\pm$2mV. Therefore a coarse ramp is generated from 0 to 60mV, 90mV to 140mV and 160mV to 298mV. Both increasing and decreasing ramps are generated, the former to determine the gain and latter to calculate the digital calibration coefficients. Figure FC3.8 shows the ramp voltage.

For the purposes of the present work, the calibration logic was implemented by a Verilog HDL code. This code was run along with the Spectre simulation of the analog circuit of the ADC in a mixed-mode AMS simulation environment of Cadence.

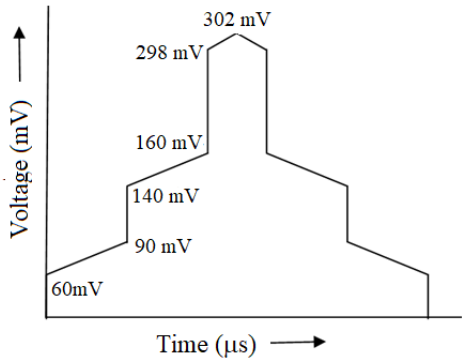

Figure FC3.8: Generation of fine and coarse ramp

3.2.1 Simulation Results

A ramp signal varying from -1.2V to 1.2V was applied to the ADC. The outputs of the ADC before and after calibration for capacitance mismatch error of up to 5% in the 3 MSB stages are shown in Figure FC3.9. Figure FC3.9 shows that the proposed algorithm effectively eliminates the errors caused by capacitance mismatches. A couple of jumps are highlighted in Figure FC3.9. Figure FC3.10 shows the INL plots of the

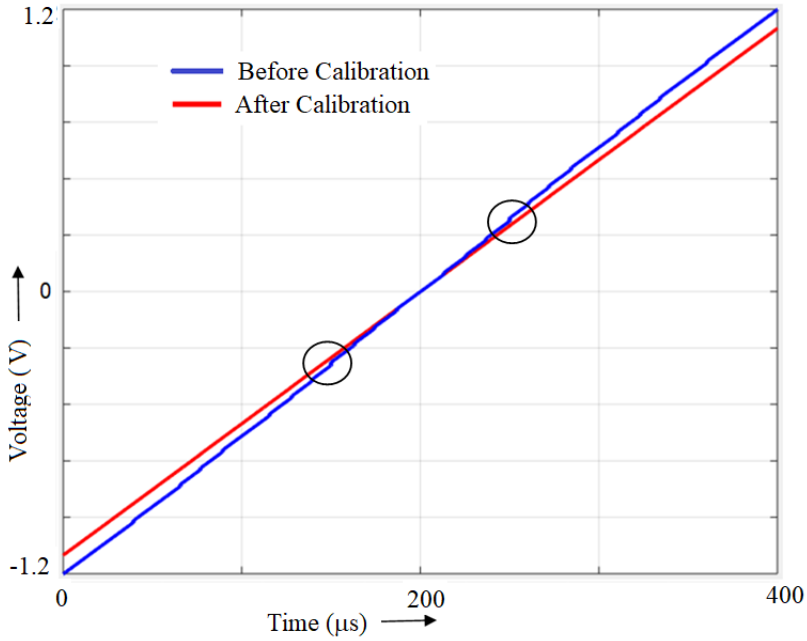

Figure FC3.9: Output transfer characteristics of the ADC before and after calibration

ADC before and after calibration. As can be seen, INL is reduced significantly after calibration. Figure FC3.11(a) and Figure FC3.11(b) show the output spectrum of a

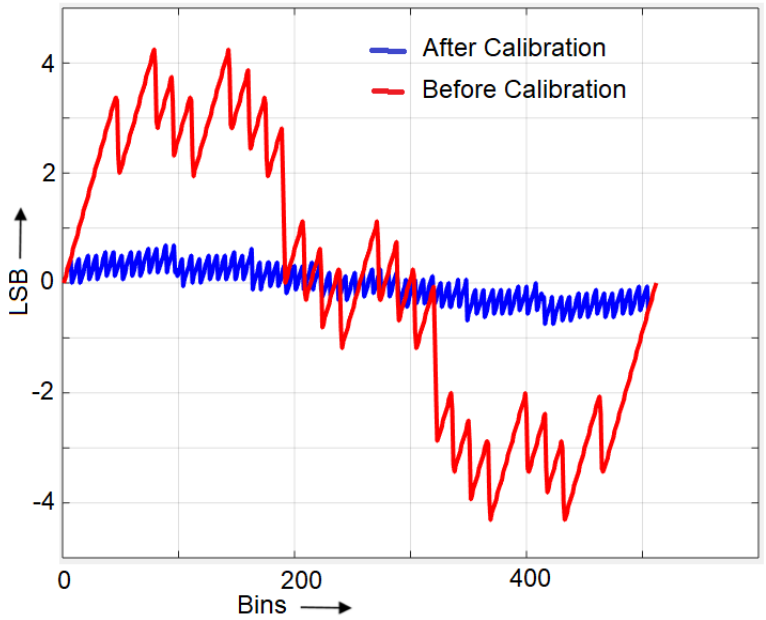

Figure FC3.10: INL of the 1.5 bits/stage pipelined ADC Before Calibration and After Calibration

1.45MHz full scale analog sine input which is sampled at 4 MSPS (N= 2048), before and after calibration. It is seen that SFDR improves from 47dB before calibration to 62 dB after calibration. The power consumed by the ADC was 110mW, out of which digital calibration hardware consumes only 76 μW. The power of the digital calibration circuit was estimated by the Cadence-Genus tool after synthesis of the Verilog HDL code to the target 180-nm CMOS SCL library. The speed of the ADC is limited by the unity gain bandwidth of the op-amp. The op-amp used in this design [22] allows the ADC to operate with maximum sampling frequency of 40MHz. For the purpose of testing the digital calibration algorithm, the input signal is being sampled at f_s = 4MSPS. The designed ADC allows an input frequency of up to $\frac{f_s}{3}$ with little SNR degradation for 2% capacitance mismatch. Figure FC3.12(a) shows the output plots for a 10-stage, 1-bit/stage ADC with capacitance mismatches of 5% in the first 3 stages, before and after calibration, simulated in Matlab. Once again, it can be seen that the post-calibration plot

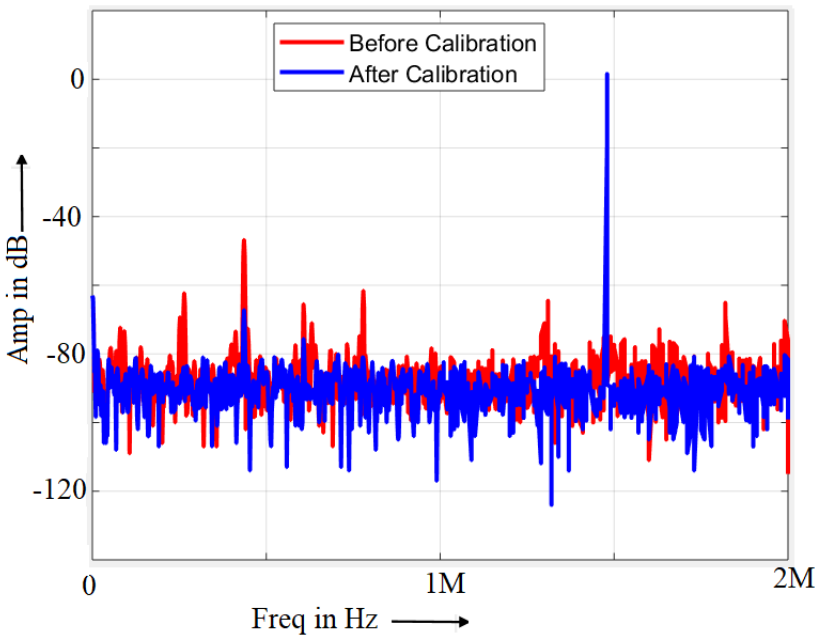

Figure FC3.11: Pipelined ADC output spectrum Before and After Calibration

eliminates the jumps due to capacitance mismatches. Figure FC3.12(b) shows the INL of the ADC after calibration, which is seen to be within ± 0.5 LSB. Similar to Figure FC3.12(a) and Figure FC3.12(b), Figure FC3.13(a) and Figure FC3.13(b) show the characteristics of a one bit/stage algorithmic ADC for a 9 bits resolution simulated in Matlab. Once again it can be seen that the proposed algorithm is effective in obtaining a linear output characteristic and an INL in an acceptable range.

Table TC3.2 summarizes the performance of the ADC.

3.3 Summary

A simple and efficient deterministic digital calibration technique is proposed to calibrate the errors due to capacitance mismatch in pipelined ADCs. The technique handles both missing codes and missing decision levels, thus reducing the INL of the ADC. The technique was demonstrated on a 10-stage, 1.5-bits/stage pipelined ADC. With upto

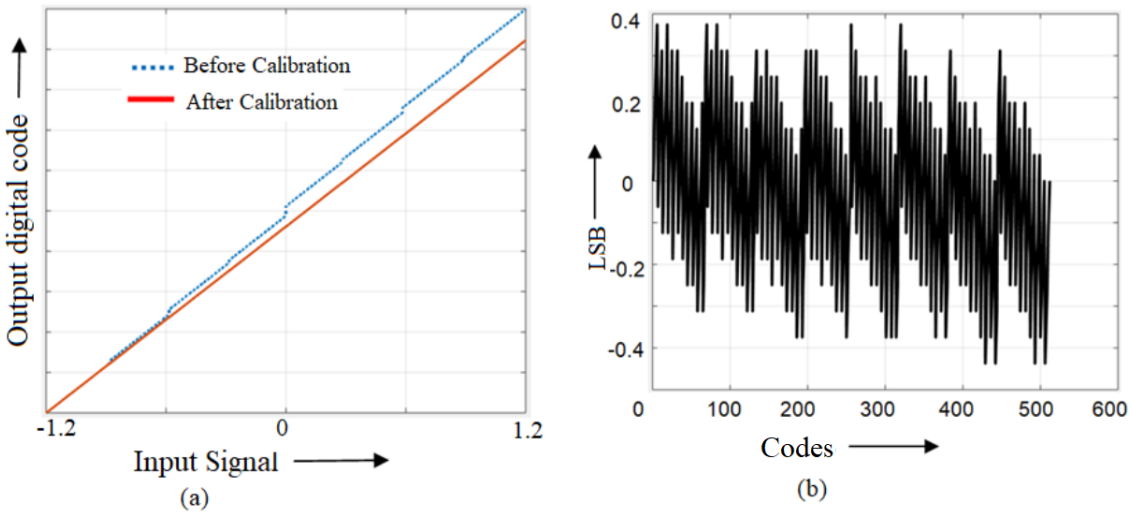

Figure FC3.12: (a) Output Characteristics of 1bit/stage pipelined ADC (b)INL of 1 bit/stage pipelined ADC after calibration

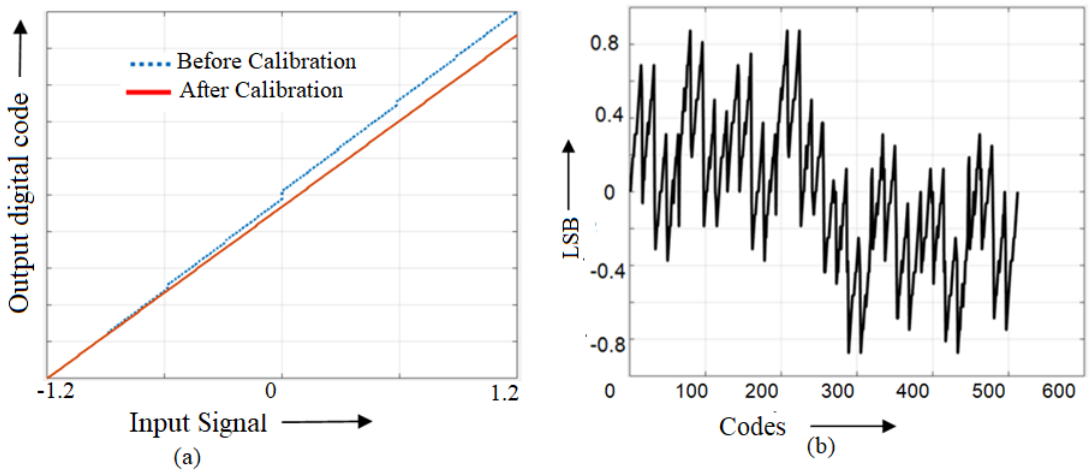

Figure FC3.13: (a) Output Characteristics of 1bit/stage algorithmic ADC (b) INL of 1 bit/stage algorithmic ADC after calibration.

Table TC3.2: Performance table of the 1.5 bits/stage ADC

Resolution	9-bits
Sampling rate	4MHz
SNR	50 dB@1.45MHz
SFDR	63dB@1.45MHz
INL	0.72LSB
DNL	0.56LSB
Input voltage swing	1.2Vp-p
Power dissipation (Without scaling currents in stages)	110mW
Predicted power dissipation by scaling consecutive stages	40mW
Process	1.8V, 180nm CMOS
Power supply	1.8V

5% capacitance mismatch in the three MSB stages, calibration reduced the INL to less than 0.8 LSB for inputs ranging from -1.2V to 1.2V. The technique can be used for algorithmic ADCs as well.

CHAPTER 4

DETERMINISTIC DIGITAL CALIBRATION TECHNIQUE FOR 1.5 BITS/STAGE PIPELINED AND ALGORITHMIC ADCS WITH FINITE OP-AMP GAIN AND LARGE CAPACITANCE MISMATCHES

This chapter proposes a high speed deterministic digital technique to calibrate the errors due to capacitance mismatch and finite op-amp gain. Unlike other calibration techniques, this technique requires neither forcing the inputs of the intermediate stages being calibrated to exact voltages, nor reducing the gains of each stage to avoid saturation of output digital codes.The advantages of the deterministic digital calibration technique presented are as follows: It can handle both stage gains of greater than 2 and less than 2; the pipeline is neither interrupted nor externally controlled during calibration; it can be used to correct finite op-amp gain error as well as capacitance mismatch. The calibration technique can be applied to any 1.5 bits/stage pipelined ADC if there is an access to the digital bits from the pipeline stage. A new technique to calibrate algorithmic ADCs is also proposed in this chapter. A 1.5 bits/stage, 10 stages, 9 bits pipeline ADC with the three most significant stages calibrated is demonstrated in this chapter. For a 10% mismatch in capacitances in the 3 MSB stages of the pipelined ADC, the calibration technique improved SNDR by more than 20dB and SFDR by around 27dB. The technique can also be slightly modified to calibrate algorithmic ADCs. For a 7%

mismatch in capacitances of the algorithmic ADC, the proposed calibration technique improved SNDR by 18dB and SFDR by around 28dB.

4.1 Proposed Algorithm to calibrate pipelined ADCs

The proposed algorithm has been demonstrated with the help of 1.5 bits/stage, 10 stage pipelined ADC which includes errors due to finite op-amp gain and capacitance mismatch. The input analog voltage ranges from –1.2 V to 1.2 V. A decreased gain leads to missing codes while an increased gain leads to missing decision levels. This is shown in Figure FC3.1 assuming a capacitance mismatch in the first stage and an ideal backend ADC consisting of the remaining stages. The above-mentioned errors can be corrected by calculating the capacitance mismatch and gain mismatch of all the stages to be calibrated. To demonstrate the proposed algorithm, we calibrate 3 MSB stages of the pipeline ADC assuming an ideal backend ADC (stages 4-10). For calibrating 3 MSB stages, the algorithm requires calibration of the 3^{rd} stage first, then the second stage and finally the first stage, because the calibration coefficients of the 3^{rd} stage are used to calculate the calibration coefficients of the second stage, and so on.

We begin with the working of the multiply-by-2 (MX2) stage of the ADC. Figure FC4.1 shows a fully-differential MX2 circuit. During the sampling phase ($\phi 2$ high), the capacitors C_1 and C_2 are charged to an voltage equal to input voltage minus the offset voltage since the op-amp is connected in unity gain configuration. This offset voltage will be cancelled in the following amplification phase ($\phi 1$ high) as the circuit performs autozeroing. The disturbances caused by the charge injection due to switches, appear as common-mode signals and are rejected by the fully balanced differential amplifier to a great extent . During the amplification phase ($\phi 1$ high), C_1 becomes the feedback capacitor and C_2, the input(sampling) capacitor. Due to autozeroing, the contribution of offset voltage of the op-amp is very insignificant and can be neglected. The residue

output, $V_{res,n}$ of any n^{th} MX2 stage at the end of the amplification phase can be written as shown in Eqn 4.1.

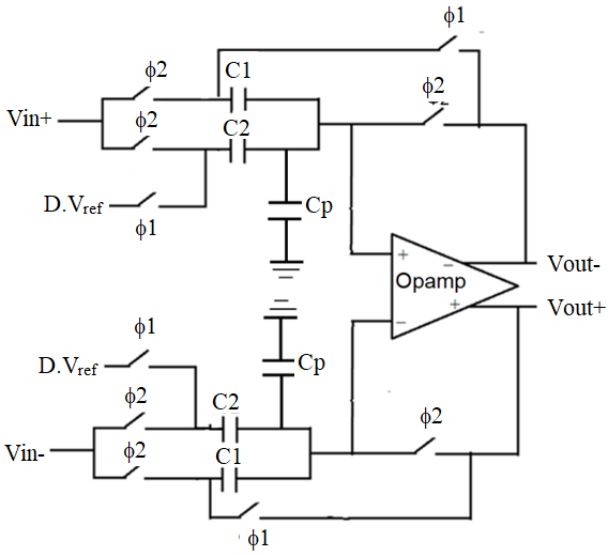

Figure FC4.1: Multiply-by-2 DAC (MX2) circuit

$$V_{res,n} = \frac{(C_1(n)+C_2(n)).V_{in}(n) - V_{ref}.C_2(n).D_n}{C_1(n) + \frac{C_2(n)+C_p(n)+C_1(n)}{A(n)}} \qquad \text{(Eqn 4.1)}$$

Where C_1(n) is the feedback capacitor of the n^{th} stage, C_2(n) is the sampling capacitor, V_{in}(n) is the analog input to the n^{th} stage, V_{ref} is the reference voltage, C_p(n) is the parasitic capacitance, A(n) is the finite open loop gain and D_n is the digital output code:

$$\begin{aligned} D_n &= 1\ ;\ if\ V_{res,n-1} > \frac{V_{ref}}{4}, \\ &= -1\ ;if\ V_{res,n-1} < -\frac{V_{ref}}{4}, \\ &= 0\ ;Otherwise. \end{aligned} \qquad \text{(Eqn 4.2)}$$

Eqn 4.1 can be re-written as,

$$V_{res,n} = \alpha_n.V_{in,n} - V_{ref}.\beta_n.D_n = \alpha_n.V_{res,n-1} - V_{ref}.\beta_n.D_n \qquad \text{(Eqn 4.3)}$$

where

$$\alpha_n = \frac{(C_2(n)+C_1(n))}{C_1(n)+\frac{C_2(n)+C_p(n)+C_1(n)}{A(n)}},$$
$$\beta_n = \frac{C_2(n)}{C_1(n)+\frac{C_2(n)+C_p(n)+C_1(n)}{A(n)}},$$

In an ideal case, the open loop gain of the op-amp is infinite, C_p is zero, and C1 = C2, so that α=2 and β = 1. In an actual circuit, due to capacitance mismatch and finite op-amp gain, α and β differ from their ideal values. This causes discontinuities and changes in slope in the transfer characteristics, as shown in Figure FC3.1. The goal of deterministic calibration techniques is to measure the magnitude of these discontinuities, to determine the actual values of α and β from these and use these actual values to obtain a calibrated digital output. In all such techniques, the operations for calculating α and β are performed on the digital output, and their values are stored in a fixed point format.

Figure FC4.2 shows the block diagram of the 1.5-bits/stage, 10-stage pipelined ADC, with the relevant symbols indicated [23]. The 3 MSB stages are assumed to be non-ideal, and the rest of the stages are assumed ideal. Equations inside the boxes for each stage are reproductions of Eqn 4.3. The digital outputs shown (the D_i's) are the raw outputs of the 1.5-bit stage that can take three different values as shown in Eqn 4.2. Note that the analog output of the n^{th} stage ($V_{res,n}$) converts to (10-n) raw digital output codes. The residue voltage of any stage can vary from $-V_{ref}$ to $+V_{ref}$.

For example, the digital equivalent of the residue voltage of the 3^{rd} stage can be written as in Eqn 4.4

$$D_{4-10} = D_4.2^{-1} + D_5.2^{-2} + D_6.2^{-3} + D_7.2^{-4} + D_8.2^{-5} + D_9.2^{-6} + D_{10}.2^{-7}$$

(Eqn 4.4)

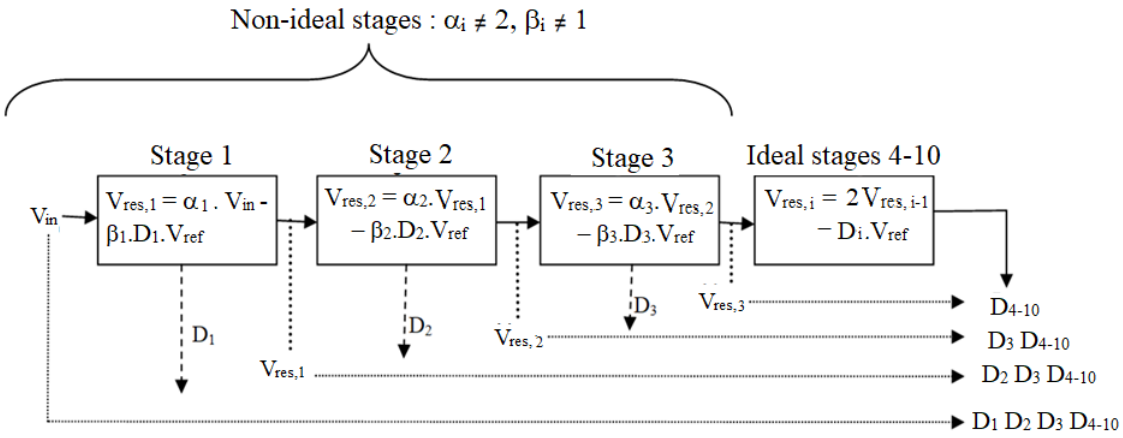

Figure FC4.2: Block diagram of the pipelined ADC, with the relevant equations and symbols

where D_i can take the values –1, 0 or 1. The analog equivalent of D_{4-10} is V_{ref} if all the codes are 1, and it is $-V_{ref}$ if all the codes are –1. We observe that non-idealities in the n^{th} stage cause discontinuities in the transfer characteristics at input voltages for which the n^{th} digital output (D_n) changes from 0 to +1 [3]. This discontinuity can be measured by monitoring the digital output of the ADC as the input is swept from 0 to $\frac{V_{ref}}{4}$, and from this, α and β can be calculated, as discussed below.

A finely incremented voltage ramp is applied to the input (V_{in}) of the ADC [23].

4.1.1 Finding α_3 and β_3

When V_{in} is close to $\frac{V_{ref}}{16}$, the input of stage 3 ($V_{res,2}$) changes from $\frac{V_{ref}}{4}$ -ΔV to $\frac{V_{ref}}{4}$+ΔV, and D_3 transitions from 0 to +1 (cf.Figure FC4.2), and non-idealities in stage 3 cause a discontinuity in the digital output (cf. Figure FC4.3). Due to redundancy in each pipeline stage, the voltage ΔV ≈ $\frac{1}{2}$ LSB. The voltage, ΔV, has been neglected further for all calculations. From Eqn 4.3, for $V_{res,2} \approx \frac{V_{ref}}{4}$ and $D_3 = 0$, the output of the 3^{rd} stage, $V_{res,3}(D_3{=}0)$, is given by Eqn 4.5

$$V_{res,3}(D_3 = 0) = \alpha_3 \frac{V_{ref}}{4} \qquad \text{(Eqn 4.5)}$$

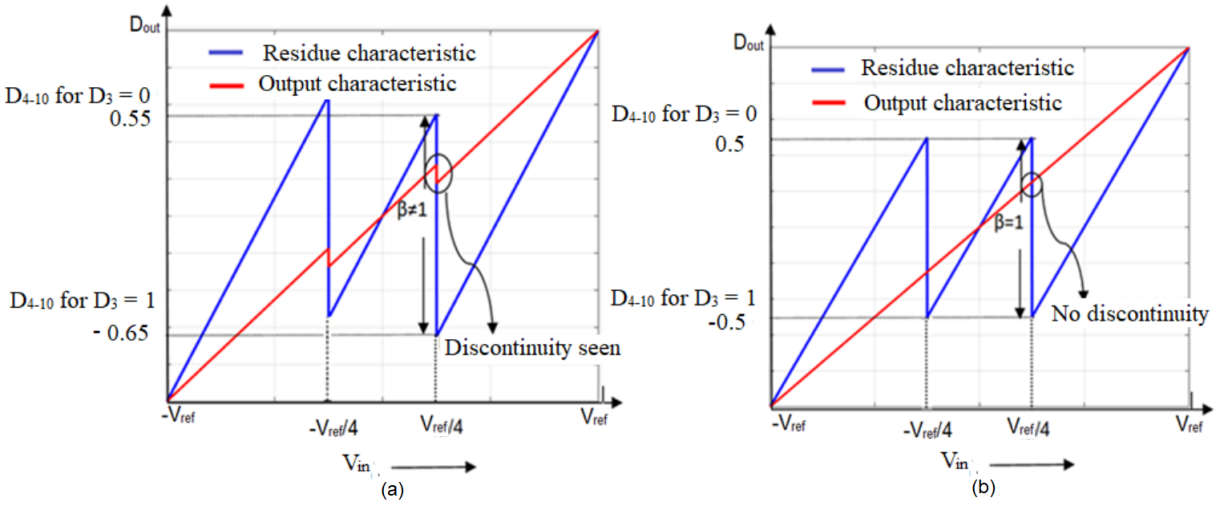

Figure FC4.3: (a) Details of discontinuity in digital output (Stages 3-10) for non-idealities in stage 3, (b) Digital output (Stages 3-10) for an ideal 3^{rd} stage pipelined ADC

For $V_{res,2} \approx \frac{V_{ref}}{4}$ and $D_3 = 1$, the output of the 3^{rd} stage, $V_{res3}(D_3{=}1)$, is given by Eqn 4.6

$$V_{res,3}(D_3 = 1) = \alpha_3 \frac{V_{ref}}{4} - \beta_3 . V_{ref} \qquad \text{(Eqn 4.6)}$$

As stages 4-10 are assumed ideal, they accurately convert the analog voltages of Eqn 4.5 and Eqn 4.6 to digital outputs (D_{4-10}), which are obtained by dividing these equations by V_{ref}. These are as shown in Eqn 4.7 and Eqn 4.8.

$$D_{4-10}(D_3 = 0) = \frac{\alpha_3}{4}. \qquad \text{(Eqn 4.7)}$$

$$D_{4-10}(D_3 = 1) = \frac{\alpha_3}{4} - \beta_3. \qquad \text{(Eqn 4.8)}$$

As indicated in Figure FC4.3, $D_{4-10}(D_3{=}0)$ and $D_{4-10}(D_3{=}1)$ are measured at the point where a discontinuity is detected in D_{out}, for the input voltage around $\frac{V_{ref}}{4}$. Solving the

last two equations, we get Eqn 4.9 and Eqn 4.10.

$$\beta_3 = D_{4-10}(D_3 = 0) – D_{4-10}(D_3 = 1) \quad \text{(Eqn 4.9)}$$

$$\alpha_3 = 4.D_{4-10}(D_3 = 0) \quad \text{(Eqn 4.10)}$$

and thus α_3 and β_3 are determined.

4.1.2 Finding α_2 and β_2

As $V_{res,2}$ passes through the non-ideal stage 3, the conversion of $V_{res,2}$ to the digital output is not accurate. Thus, unlike Eqn 4.9 and Eqn 4.10, α_2 and β_2 cannot be directly determined from the measured residue digital outputs. We write the equations Eqn 4.11 and Eqn 4.12 to determine α_2 and β_2 [23]. Writing Eqn 4.3 for stages 3 and 2, respectively:

$$V_{res,3} = \alpha_3.V_{res,2} – V_{ref}.\beta_3.D_3 \quad \text{(Eqn 4.11)}$$

$$V_{res,2} = \alpha_2.V_{res,1} – V_{ref}.\beta_2.D_2 \quad \text{(Eqn 4.12)}$$

Substituting Eqn 4.12 in Eqn 4.11, we get Eqn 4.13

$$V_{res,3} = \alpha_3(\alpha_2.V_{res,1} – V_{ref}.\beta_2.D_2) – V_{ref}.\beta_3.D_3 \quad \text{(Eqn 4.13)}$$

When V_{in} is close to $\frac{V_{ref}}{8}$, the input of stage 2 ($V_{res,1}$) crosses $\frac{V_{ref}}{4}$, and D_2 transitions from 0 to +1 (cf. Figure FC4.2 and Figure FC4.3). Non-idealities in stage 2 cause a discontinuity in the digital output at this point. Similar to Eqn 4.5 and Eqn 4.6, we

write Eqn 4.14 and Eqn 4.15 for D_2=0 and D_2=1 respectively.

$$V_{res,3}(D_2 = 0) = \alpha_3(\alpha_2.\frac{V_{ref}}{4})–V_{ref}.\beta_3.D_3(D_2 = 0) \quad \text{(Eqn 4.14)}$$

$$V_{res,3}(D_2 = 1) = \alpha_3(\alpha_2.\frac{V_{ref}}{4} - V_{ref}.\beta_2)–V_{ref}.\beta_3.D_3(D_2 = 1) \quad \text{(Eqn 4.15)}$$

where $D_3(D_2$=0) and $D_3(D_2$=1) are the stage 3 digital outputs when V_{in} crosses $\frac{V_{ref}}{8}$ for D_2=0 and D_2=1, respectively. As with Eqn 4.7 and Eqn 4.8, the conversion of $V_{res,3}$ to digital outputs (D_{4-10}) is accurate. Hence, dividing the last two equations by V_{ref}, we get these digital outputs which can be measured where D_2 makes a transition from 0 to 1. This is as shown in Eqn 4.16 and Eqn 4.17.

$$D_{4-10}(D_2 = 0) = \frac{\alpha_3.\alpha_2}{4}–.\beta_3.D_3(D_2 = 0) \quad \text{(Eqn 4.16)}$$

$$D_{4-10}(D_2 = 1) = \frac{\alpha_3.\alpha_2}{4} - \alpha_3.\beta 2–.\beta_3.D_3(D_2 = 1) \quad \text{(Eqn 4.17)}$$

The above two equations can be solved to determine α_2 and β_2, as everything else in the equations is known.

4.1.3 Finding α_1 and β_1

The analysis done above for α_2 and β_2 can be repeated for stage 1, with V_{in} crossing $\frac{V_{ref}}{4}$, and D_1 transitioning from 0 to 1, to finally obtain the equations as shown in Eqn 4.18 .

$$V_{res,3} = \alpha_3.\alpha_2.\alpha_1.V_{in}–V_{ref}\alpha_3.\alpha_2.\beta_1.D_1–V_{ref}.\alpha_3.\beta_2.D_2–V_{ref}.\beta_3.D_3 \quad \text{(Eqn 4.18)}$$

For $V_{in} = \frac{V_{ref}}{4}$, with D_1 transitioning from 0 to 1, we obtain Eqn 4.19 and Eqn 4.20.

$$D_{4-10}(D_1 = 0) = \frac{\alpha_3.\alpha_2.\alpha_1}{4} - \alpha_3.\beta_2 D_2(D_1 = 0) - \beta_3.D_3(D_1 = 0) \qquad \text{(Eqn 4.19)}$$

$$D_{4-10}(D_1 = 1) = \frac{\alpha_3.\alpha_2.\alpha_1}{4} - \alpha_3.\alpha_2.\beta_1 - \alpha_3.\beta_2 D_2(D_1 = 1) - \beta_3.D_3(D_1 = 1) \qquad \text{(Eqn 4.20)}$$

where $D_2(D_1$=0) and $D_2(D_1$=1) are the stage 2 digital outputs when V_{in} crosses $\frac{V_{ref}}{4}$ for D_1=0 and D_1=1, respectively. The last two equations can be solved to determine α_1 and β_1, as everything else in the equations is known.

This marks the end of the calibration phase. During normal operation of the ADC, the values of α_i and β_i (for i=1,2,3) obtained above are used to calculate the calibrated digital output as follows. For an analog input V_{in}, let the corrected (calibrated) digital output be D_{out}. As $V_{res,3}$ yields an accurate digital output D_{4-10} (cf. Figure FC4.2), we divide Eqn 4.18 by V_{ref} to write Eqn 4.21 .

$$D_{4-10} = \alpha_3.\alpha_2.\alpha_1.D_{out} - \alpha_3.\alpha_2.\beta_1.D_1 - \alpha_3.\beta_2 D_2 - \beta_3.D_3 \qquad \text{(Eqn 4.21)}$$

Solving for D_{out}, we get Eqn 4.22

$$D_{out} = \frac{\beta_1}{\alpha_1}.D_1 + \frac{\beta_2}{\alpha_1.\alpha_2}.D_2 + \frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}.D_3 + \frac{D_{4-10}}{\alpha_1.\alpha_2.\alpha_3} \qquad \text{(Eqn 4.22)}$$

where $D_1D_2D_3D_{4-10}$ are the ADC output codes for the input V_{in}. β_1, β_2, β_2 are the capacitance mismatches of stages 1, 2 and 3, respectively, and α_1, α_2, α_2 are gain mismatches. If n stages are calibrated, then Eqn 4.22 can be generalized as shown in

Eqn 4.23

$$D_{out} = \frac{\beta_1}{\alpha_1}.D_1 + \frac{\beta_2}{\alpha_1.\alpha_2}.D_2 + \frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}.D_3 + ... + \frac{\beta_n}{\alpha_1.\alpha_2....\alpha_n}.D_n + \frac{DBE_{n-10}}{\alpha_1.\alpha_2....\alpha_n} \quad \text{(Eqn 4.23)}$$

Note that in case of an ideal ADC, $\frac{\beta_1}{\alpha_1} = 2^{-1}$, $\frac{\beta_1}{\alpha_1.\alpha_2} = 2^{-2}$ and so on.

The proposed algorithm works well assuming there are no comparator offsets. However, finite comparator offsets does not introduce non-linearity in the output characteristics of the ADC but changes the slope of the output characteristics of the ADC thus limiting the input signal range. Simulation results shows that a comparator offset of 5mV in the first stage reduces the signal range by 2%.

4.2 Proposed Algorithm to calibrate algorithmic ADCs

The method described above is modified slightly to calibrate algorithmic ADCs, because unlike pipelined ADCs, algorithmic ADCs do not have ideal backend stages [24]. As an algorithmic ADC has only one stage, calibration requires the calculation of α and β of this one stage. In case of an n bit algorithmic ADC, therefore, we say that α_1 = α_2 =α_3 =. . . =α_n =α and β_1 = β_2 =β_3 =. . . =β_n= β. Once α and β are calculated, the weights with which the digital output has to be multiplied to get the calibrated digital output (D_{out}) can be calculated from Eqn 4.22 as $\frac{\beta}{\alpha}$, $\frac{\beta}{\alpha^2}$,$\frac{\beta}{\alpha^3}$, $\frac{\beta}{\alpha^4}$ and so on, and D_{out} can be written as shown in Eqn 4.24 [24].

$$D_{out} = D_1.\frac{\beta}{\alpha} + D_2.\frac{\beta}{\alpha^2} + D_3.\frac{\beta}{\alpha^3} + D_4.\frac{\beta}{\alpha^4} + D_5.\frac{\beta}{\alpha^5}.D_6.\frac{\beta}{\alpha^6} + D_7.\frac{\beta}{\alpha^7} + D_8.\frac{\beta}{\alpha^8} + D_9.\frac{\beta}{\alpha^9} + D_{10}.\frac{\beta}{\alpha^{10}} \quad \text{(Eqn 4.24)}$$

α and β are calculated as follows. An analog voltage $\frac{V_{ref}}{4}$ is input to the algorithmic ADC with the MSB forced to 0. After 9 clock cycles, we get the 10 digital output codes from the algorithmic ADC with the output of the most significant MX2 forced to 0. The calibrated digital output D_{out0} can be obtained from the digital outputs using Eqn 4.24.

Now a voltage $\frac{V_{ref}}{4}$ is input with the MSB forced to 1. After 9 clock cycles, we get the 10 digital outputs with the output of the most significant MX2 forced to 1. The calibrated digital output D_{out1} can be obtained from the raw bits using Eqn 4.24. As D_{out} is the calibrated output, the equations obtained for D_{out} by forcing the MSB to 1 and to 0 will be equal, that is, D_{out0}= D_{out1}. β will get cancelled in this equation, thus yielding a polynomial equation for α, which can be solved numerically. We used the Newton Raphson method to solve it, with an initial estimate of $\alpha = 2$. Once α is found, β can be calculated using the equation for D_{out0} or D_{out1}.This completes the calibration phase.

During normal operation of the ADC, Eqn 4.24 is used to obtain the calibrated digital output from the raw digital output codes. Li [25] proposed ratio independent algorithmic ADC which takes multiple phases of the clock for conversion whereas the proposed technique performs conversion in just 2 phases of the clock. Erdogan [14] and [24] calibrates the ADC using LMS algorithm which requires thousands of clock cycles to converge. The proposed algorithm calibrates the ADC using Newton Raphson method which converges in just 5 cycles [26]. Figure FC4.4 shows the rate of convergence of α for a capacitance mismatch of 6.25%.

4.3 Circuit Implementation

The circuit was implemented in Semiconductor Complex Limited, India (SCL)'s 180-nm CMOS technology, and was simulated in Cadence-Spectre. As mentioned in

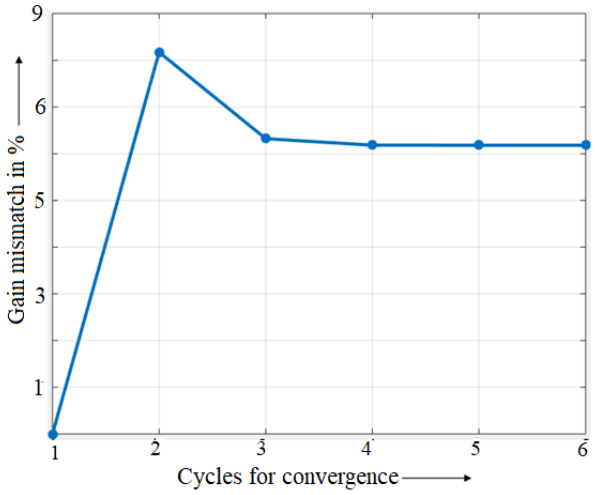

Figure FC4.4: Convergence of α for capacitance mismatch of 6.25%.

section. 4.1, a 1.5 bit/stage, 10-stage pipelined ADC was implemented as a test case. The 1.5-bit flash ADC in each stage was designed with Strong-ARM latch comparators [19]. The reference voltages ($\pm\frac{V_{ref}}{4}$) for the comparator were generated using switched-capacitor voltage divider circuit [20]. The circuit of the multiply-by-2 DAC (MX2) was discussed in section. 4.1 (see Figure FC4.1). The op-amp in the MX2 was a two-stage op-amp with a telescopic cascode first-stage followed by a push pull common source stage, and common-mode feedback for each stage. The op-amp was designed to provide a gain of 87dB, a unity-gain frequency of 500MHz, and a phase margin of 56^o over all process corners [Appendix :A.1].The first stage of the pipelined ADC was designed to work as a differential sample and hold amplifier (SHA) during normal operation and as a ramp-generating switched-capacitor integrator (SCI) during calibration. This is explained in detail in Chapter 3 (cf. Figure FC3.7).

Note that MX2 stages used in pipelined ADC works in a ping-pong fashion, sampling the input at $\phi 2$ ($\phi 1$) and producing a residue output at $\phi 1$ ($\phi 2$) which in turn is sampled by the next MX2 stage at $\phi 1$ ($\phi 2$). Algorithmic ADC cannot work in a ping pong fashion as it consists of only one MX2 stage and therefore the residue output of the MX2 stage should be held constant until it is processed by the same MX2 stage again.

This can be done using the same SHA that is used to sample the input signal. Unlike the SHA for the pipelined ADC, a conventional SHA can be used for an algorithmic ADC since ramp generation is not required to calibrate algorithmic ADC. This is as shown in Figure FC4.5. During calibration, the switches S1 and S2 connect the input of the SHA to V_{cal} during the first clock cycle to sample the calibration signal $\frac{V_{ref}}{4}$. During the rest of the 9 cycles, the input of the SHA is connected to $MX2$ which samples the residue output of the MX2 stage. During the normal operation of the ADC, the input of the SHA is connected to V_{in} during the first clock cycle to sample the input signal. During the rest of the 9 cycles, the input of the SHA is connected to $MX2$ which samples the residue output of the MX2 stage.

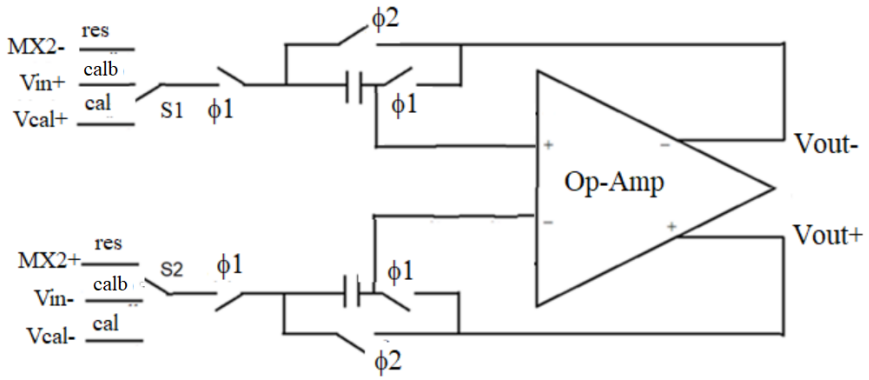

Figure FC4.5: Conventional sample and hold amplifier for an algorithmic ADC.

4.4 Simulation results

The pipelined ADC and algorithmic ADC along with the calibration scheme were implemented in a 180-nm CMOS technology of Semiconductor Complex Limited, India (SCL), and were simulated in Cadence-AMS environment. The simulation results [Bode plots] of each stage of the op-amp are as shown in [22]. The op-amp has also been simulated across the process corners and results are promising. SHA built using this op-amp shows a minimal overshot with the final value of the output settling with

an error of 60μV for a full scale input when operated at 40 MHz [22]. The representative system level timing diagram of the pipeline ADC during calibration is as shown in Figure FC4.6.

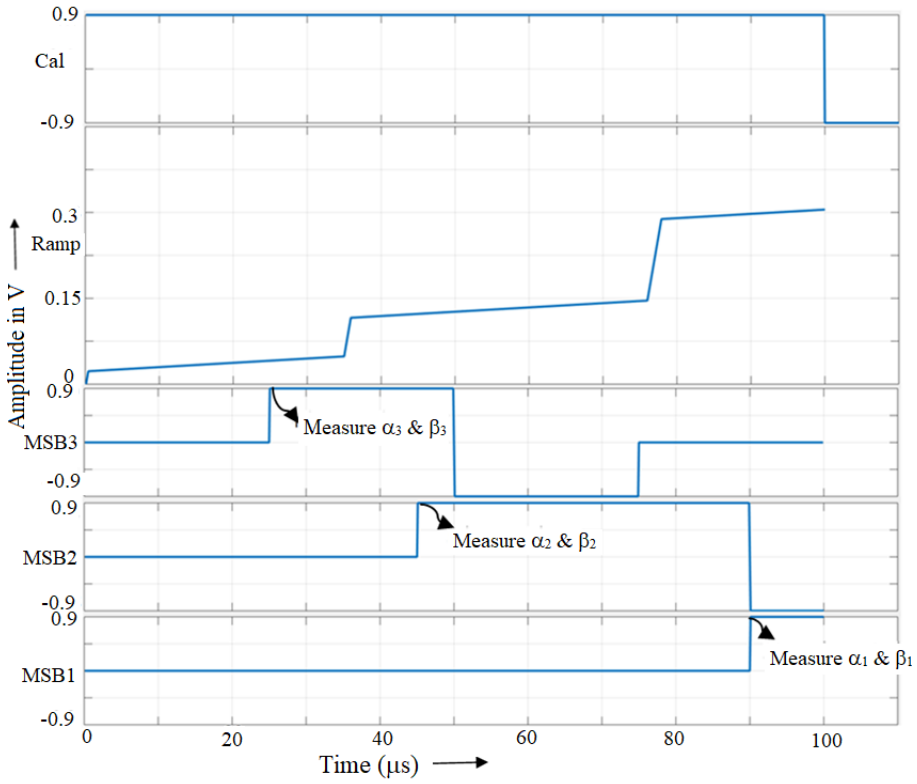

Figure FC4.6: Representative system level timing diagram of the piprline ADC during calibration

During Calibration, *Cal* signal goes high indicating the commencement of the calibration. The output of the SHA starts off with a coarse ramp and alternates between fine ramp and coarse ramp during the period of calibration. Measurement of the calibration coefficients takes place during the fine ramp while coarse ramp is to speed up the calibration process. The calibration coefficients α_n and β_n are calculated when the n^{th} digital code changes from 0 to +1. This is as shown in Figure FC4.6. α_n and β_n are stored and used later during the normal operation of the ADC to obtain calibrated digital output code.

The pipelined ADC was designed to work for inputs ranging from 1.2V to -1.2V. To test the digital calibration technique, a ramp varying from 1.2V to -1.2V was fed to the 4 MSPS pipelined ADC where the capacitance mismatches can vary upto 10%. Note that the proposed digital calibration logic can be implemented with pipelined ADCs of much higher speed as the op-amp limits the speed of the pipelined ADC and not the digital hardware used for calibration. The raw digital output and the calibrated digital output were input to an ideal DAC to obtain an analog output (for ease of visualization). The transfer characteristics of the pipelined ADC before and after calibration are shown in Figure FC4.7. The plots demonstrate that the proposed calibration technique eliminates the discontinuities caused by capacitance mismatch and gain errors and produces a linear, rail-to-rail output. A couple of discontinuities are highlighted in Figure FC4.7a.

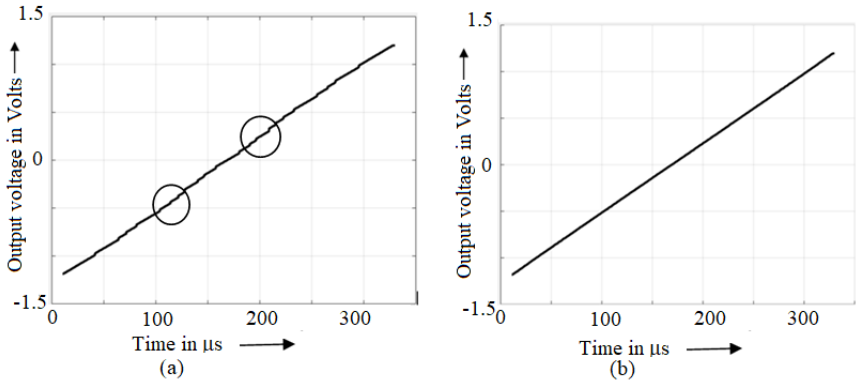

Figure FC4.7: Output transfer characteristics of the pipeline ADC (a) before calibration and (b) after calibration

The output transfer characteristics of the algorithmic ADC before and after calibration are shown in Figure FC4.8. Once again, it can be seen that the proposed technique eliminates the capacitance mismatch and gain errors to produce a linear, rail-to-rail output. A couple of discontinuities are highlighted in Figure FC4.8a.

The integral non-linearity (INL) and differential non-linearity (DNL) of the pipelined ADC were calculated using the simulation results from Cadence, as follows. Up to 10%

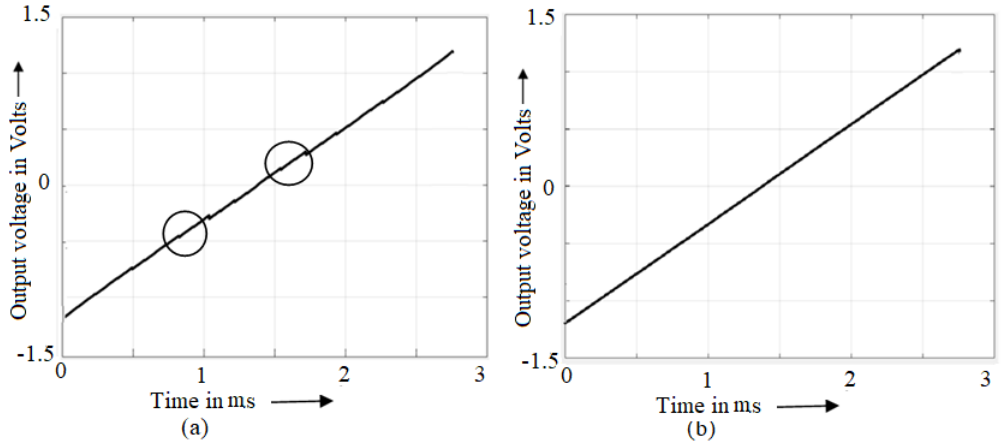

Figure FC4.8: Output transfer characteristics of the algorithmic ADC (a) before calibration and (b) after calibration

mismatch was introduced in the capacitance values of the first 3 stages. An input ramp from -1.2V to 1.2V was fed to the input. The output data was processed in Matlab to extract the INL and DNL of the ADC. The INL plots before and after calibration of the pipelined ADC is shown in Figure FC4.9. The plots show that the maximum INL is reduced to less than 0.7 LSB, after calibration. Figure FC4.10 shows the INL plot before and after calibration for a 9 bit resolution algorithmic ADC, with a 7% mismatch in the capacitance values. It is observed that the INL of the ADC is reduced to 0.64 LSB after calibration. Figure FC4.11 shows the output spectrum (N=2048) of a 1.2MHz full

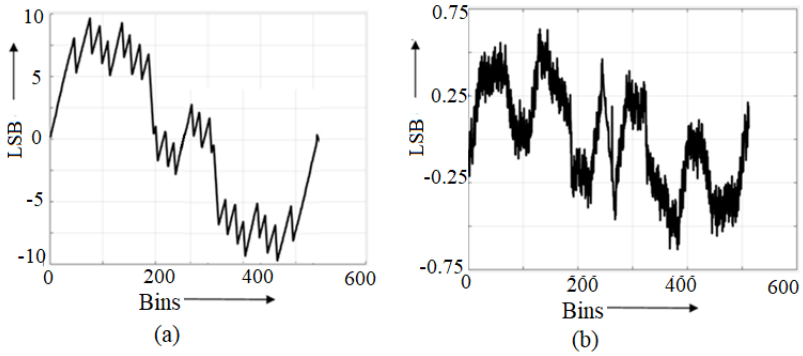

Figure FC4.9: INL of pipelined ADC. (a) Before calibration, and (b) after calibration.

scale analog sine input which is sampled at 4 MSPS by the pipelined ADC, before and

after calibration. It is seen that SFDR improves from 35dB before calibration to 62 dB after calibration.

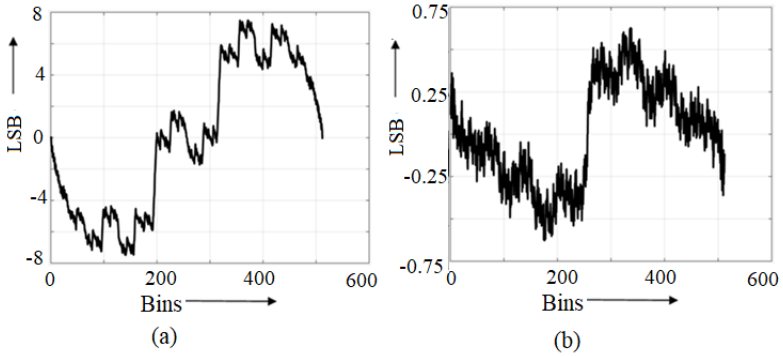

Figure FC4.10: INL of algorithmic ADC for 9 bits resolution. (a) Before calibration, and (b) after calibration

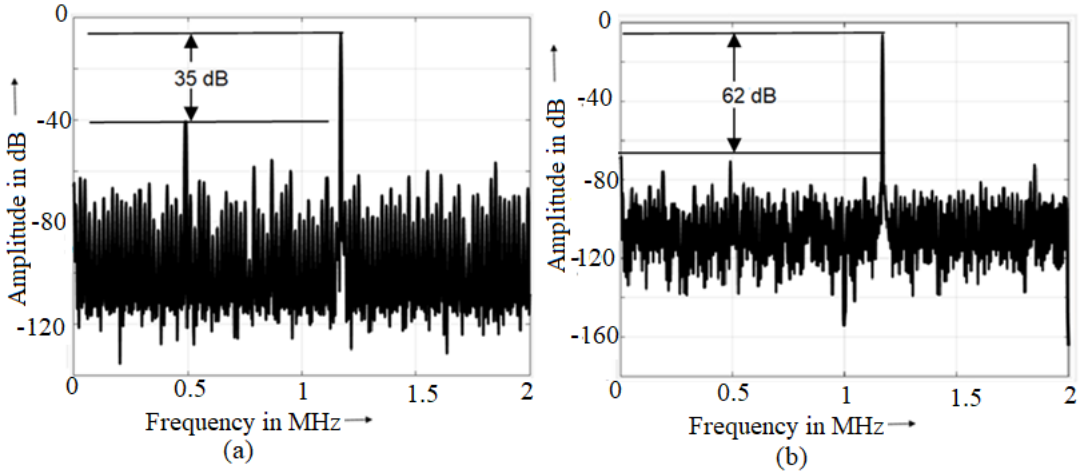

Figure FC4.11: Pipelined ADC output spectrum (a) before calibration and (b) after calibration

Figure FC4.12 shows the output spectrum (N=2048) of the 120 KHz full scale analog sine input where each sample is sampled at 400 KSPS by the algorithmic ADC, before and after calibration. It is seen that SFDR improves by 28dB after calibration. Figures FC4.13 and FC4.14 show an improvement in SNDR and Effective Number Of Bits (ENOB) for a full scale analog input before and after calibration of the pipelined ADC and algorithmic ADC respectively. Table TC4.1 and Table TC4.2 give the per-

formance of the pipelined ADC and algorithmic ADC respectively.

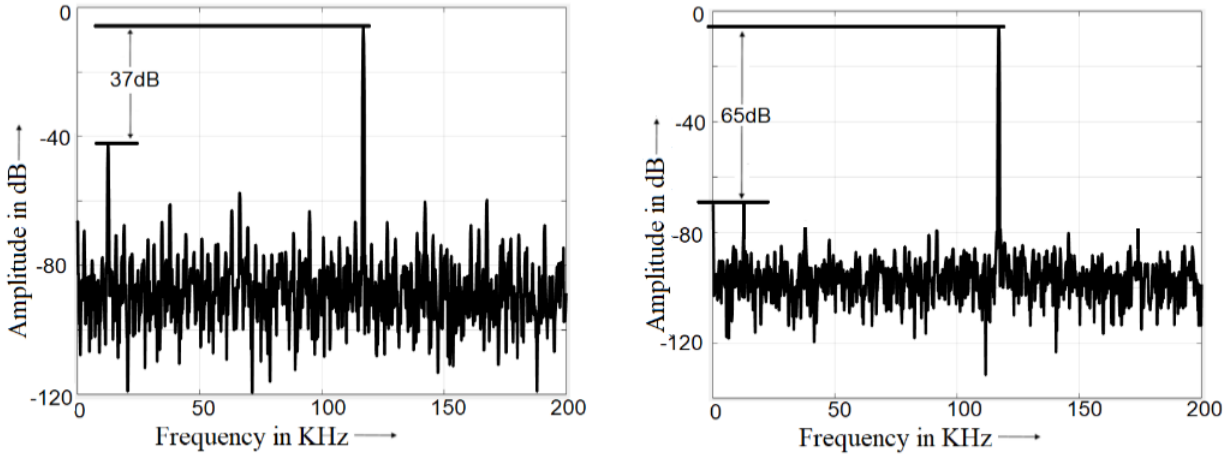

Figure FC4.12: Algorithmic ADC output spectrum for 10 bits resolution (a) before calibration and (b) after calibration

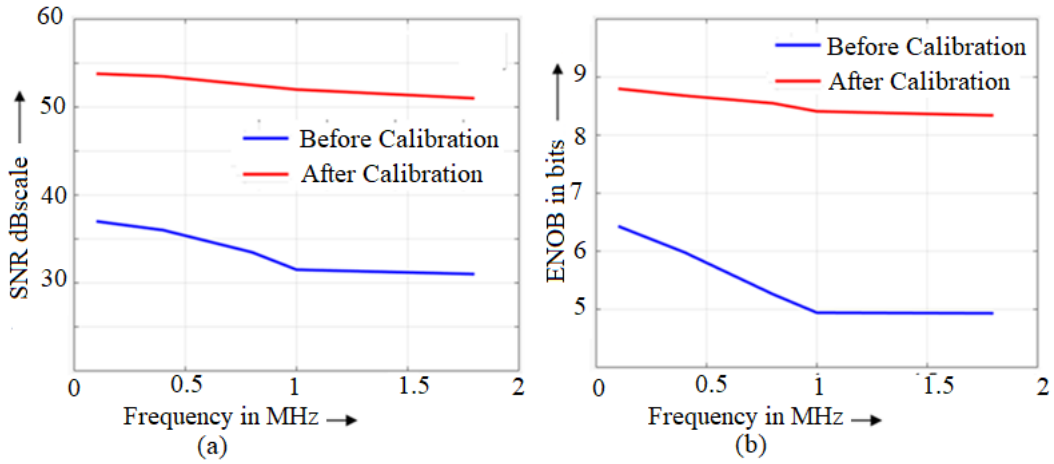

Figure FC4.13: SNDR and ENOB plot of the pipelined ADC

4.5 Conclusions

A novel deterministic digital calibration technique is proposed to correct both capacitance mismatches as well as gain errors, in 1.5-bits/stage pipelined as well as algorithmic ADCs. To illustrate the technique, a 10-stage, 1.5-bits/stage pipelined ADC,

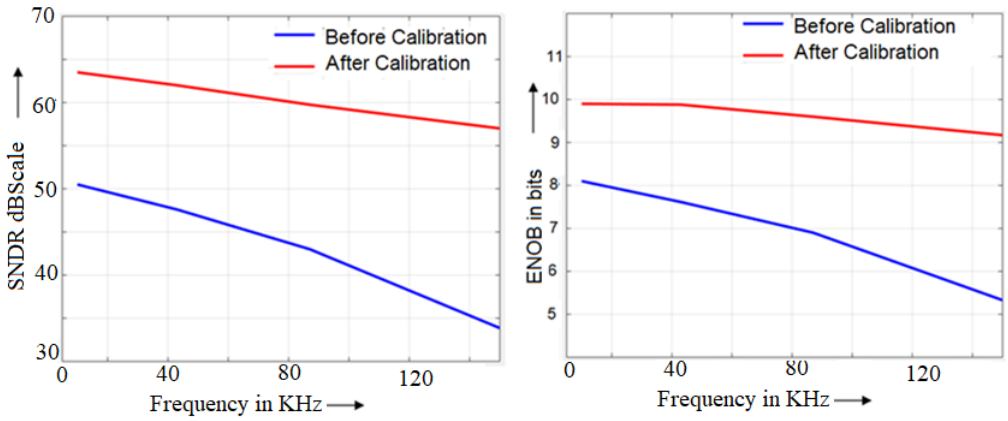

Figure FC4.14: SNDR and ENOB plot of the Algorithmic ADC for 10 bits resolution

Table TC4.1: Performance table of the 1.5 bits/stage pipelined ADC

Resolution	9-bits
Sampling rate	4MHz
SNR	51 dB@1.8MHz
SFDR	58 dB@1.8MHz
INL	0.66LSB
DNL	0.4LSB
Input voltage swing	1.2Vp-p
Power dissipation (Without scaling currents in stages)	110mW
Predicted power dissipation by scaling consecutive stages	40mW
Power dissipated by the calibration hardware synthesized using Cadence-Genus	460uW
Process	1.8V, 180nm CMOS
Power supply	1.8V

and a 1.5-bit stage algorithmic ADC were designed in a 180nm technology. With 10% mismatch in capacitance values, INL and DNL of the calibrated pipelined ADC were reduced to 0.66 LSB and 0.4 LSB, respectively, and with 7% mismatch, the INL and DNL of the calibrated algorithmic ADC were both reduced to 0.64 LSB and 0.4 LSB. A significant improvement in the SNDR, ENOB and SFDR after calibration in both pipelined as well as algorithmic ADC was also observed.

Table TC4.2: Performance table of the 1.5 bits/stage algorithmic ADC

Resolution	9/10-bits
Sampling rate	400KHz
SNR	57dB @150KHz
SFDR	63dB @150KHz
INL	0.64LSB/1.28 LSB
DNL	0.4LSB/ 0.8 LSB
Input voltage swing	1.2Vp-p
Power dissipation (SHA+MX2 stage)	19mW
Power dissipated by the calibration hardware synthesized using Cadence-Genus	3.6mW
Process	1.8V, 180nm CMOS
Power supply	1.8V

CHAPTER 5

DETERMINISTIC DIGITAL CALIBRATION OF 1.5 BITS/STAGE PIPELINED ADC BY DIRECT EXTRACTION OF CALIBRATION COEFFICIENTS

This chapter proposes a novel foreground digital calibration technique for 1.5 bits/stage pipeline ADCs to calibrate ADC errors due to capacitor mismatch and finite op-amp gain. The proposed algorithm directly extracts the weights with which the digital raw code of the pipelined ADC has to be multiplied, with minimal digital operations. Unlike other foreground calibration techniques, the proposed technique does not require forcing the inputs and outputs of the intermediate stages to be calibrated. A 1.5 bits/stage, 12 stage ADC with the four MSB stages calibrated is demonstrated. The INL and DNL of the calibrated ADC lie below 0.65 LSB and 0.3 LSB, respectively. In addition, an algorithm for the calibration of an algorithmic ADC is described.

5.1 Proposed Calibration Technique

The proposed algorithm is demonstrated with the help of a 1.5 bits/stage, 12 stage pipelined ADC where 4 MSB stages are calibrated. Each pipeline stage is made up of a multiply-by-2-amplifier (MX2) using a flip-around DAC and comparators. A detailed working of the MX2 stage along with the equations are given in Chapter 4 (cf. Figure

FC4.1).

Considering a 12 stage pipelined ADC where 4 stages are calibrated with an ideal 8-bit backend ADC, the calibrated digital output, D_{out}, can be written as shown in Eqn 5.1.

$$D_{out} = \frac{\beta_1}{\alpha_1}.D_1 + \frac{\beta_2}{\alpha_1.\alpha_2}.D_2 + \frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}.D_3 + \frac{\beta_4}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}.D_4 + \frac{D_5}{2^1.\alpha_1.\alpha_2.\alpha_3.\alpha_4} + ... + \frac{D_{12}}{2^8.\alpha_1.\alpha_2.\alpha_3.\alpha_4} \quad \text{(Eqn 5.1)}$$

Where D_i is the digital output code from the i^{th} MX2 stage, α_i and β_i are represented in fixed point format. Eqn 5.1 can be re-written as shown in Eqn 5.2.

$$D_{out} = D_1.W_1 + D_2.W_2 + D_3.W_3 + ...D_{12}.W_{12} \quad \text{(Eqn 5.2)}$$

The proposed algorithm obtains all the weights (calibration coefficients), $W_1 = \frac{\beta_1}{\alpha_1}$, $W_2 = \frac{\beta_2}{\alpha_2.\alpha_1}$, $W_3 = \frac{\beta_3}{\alpha_3.\alpha_2.\alpha_1}$, $W_4 = \frac{\beta_4}{\alpha_4.\alpha_3.\alpha_2.\alpha_1}$, $W_5 = \frac{1}{2^1.\alpha_4.\alpha_3.\alpha_2.\alpha_1}$, required to calculate the calibrated digital output with minimal mathematical manipulations. With an ideal backend ADC, the weights of the LSBs D_6 to D_{12} can be calculated by dividing the weight of D_5 by appropriate powers of 2 (right shift operation).

The algorithm requires a slow and accurate ADC like an algorithmic ADC or sigma delta ADC that gives an accurate digital output of the analog input voltage in fixed point format. An algorithmic ADC with one MX2 stage is used as reference ADC. It is required to set the gain of the 4 MSB stages of the pipeline ADC to a little greater than 2. Also, the 1^{st} stage gain of the pipeline ADC should be greater than the 2^{nd} stage gain which in turn should be greater than the 3^{rd} stage gain which should be greater than 4^{th} stage gain. This is to make sure that there are no missing codes introduced by the pipelined ADC for a slow ramp input during calibration. The high level view of the calibration algorithm is as shown in Figure FC5.1.

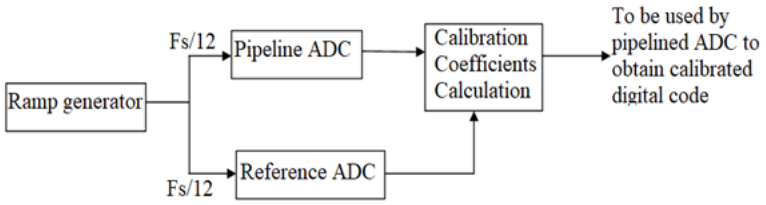

Figure FC5.1: High level view of the calibration algorithm

The proposed algorithm works as follows. A fine voltage ramp is input to the pipeline ADC and to the algorithmic ADC. The pipeline ADC has to wait until the algorithmic ADC completes conversion before it receives the next input, since both pipeline ADC and algorithmic ADC convert the same input signal during calibration. The step size of the ramp should be such that the algorithmic ADC does not encounter missing codes in between. The pipeline ADC weights W_1-W_{12} are measured during calibration and stored in digital fixed point format so that they can be used to get a calibrated digital code during the normal operation of the ADC. The calculated weights of the pipeline ADC are accurate to 12 bits.

5.1.1 Finding $W_5 - W_{12}$

When the ramp input reaches an analog value close to $\frac{V_{ref}}{32}$, the digital output code of the pipelined ADC is monitored until it produces a digital output code "0,0,0,0,1,0,0,0,0, 0,0,0" . This corresponds to a calibrated digital output of $\frac{1}{2^1.\alpha_1.\alpha_2.\alpha_3.\alpha_4}$ (Eqn 5.1). Ideally, $\frac{1}{2^1.\alpha_1.\alpha_2.\alpha_3.\alpha_4} = \frac{1}{32}$. But, due to capacitance mismatches and finite gain of the op-amp, the actual analog input that produces this output digital code differs from $\frac{V_{ref}}{32}$. The input signal for which the pipeline ADC produces the digital output of "0,0,0,0,1,0,0,0,0,0,0,0" can be accurately measured using the digital output from algorithmic ADC, D_{algo1}. Therefore, D_{algo1}, which is stored in fixed point format, gives W_5. The weights of the LSB bits W_6 - W_{12} can be obtained by dividing W_5 by 2^1, 2^2 up to 2^7 respectively. The representative circuit diagram for the generation of the weights

W_5 – W_{12} is as shown in Figure FC5.2. Each MX2 stage of the pipelined ADC outputs a digital code which can take a value of 1, 0, or -1. These are represented using 2 digital bits. The digital code '1' is represented by '10', '0' by '01' and '-1' by '00'. The MSB of the digital code of the 5^{th} stage ($D5\_MSB$) and LSB of the digital codes of the rest of the stages ($D1\_LSB$ to $D12\_LSB$ except $D5\_LSB$) are used in the calculation of W_5. The weight W_5 is calculated when the output digital code of the pipeline ADC is "0,0,0,0,1,0,0,0,0,0,0,0" each code represented using 2 bits. For $D5\_MSB$ = 1 and $D1\_LSB$ to $D12\_LSB$ except $D5\_LSB$ = 1, the corresponding 12 bits digital output from the algorithmic ADC is stored as W_5. The right shift operation on W_5 yields W_6, right shift on W_6 produces W_7 and so on [27].

5.1.2 Finding W_4

Similar to finding W_5, W_4 can be measured by monitoring the digital output of the pipelined ADC for the digital code "0,0,0,1,0,0,0,0,0,0,0,0" whose analog equivalent ($\approx \frac{V_{ref}}{16}$) is given by the algorithmic ADC in digital fixed point format, D_{algo2}. This code D_{algo2}, corresponds to the pipeline ADC calibrated digital output of $\frac{\beta_4}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}$ which is equal to W_4 (Eqn 5.1).

5.1.3 Finding W_3

W_3 can be calculated similar to W_4 and W_5. But this increases the calibration time (explained in detail in the next section). Therefore W_3 is calculated as follows. The digital output of the ADC is monitored for the output digital code "0,0,1,-1,0,0,0,0,0,0,0,0". This corresponds to a pipeline ADC digital calibrated output of $\frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}$ - $\frac{\beta_4}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}$ (Eqn 5.1) which ideally corresponds to an analog voltage of $\frac{V_{ref}}{16}$. The digital equivalent of the actual analog input voltage, D_{algo3}, is obtained using algorithmic ADC. The weight, W_3, can be calculated by adding D_{algo2} with D_{algo3} which corresponds to digital

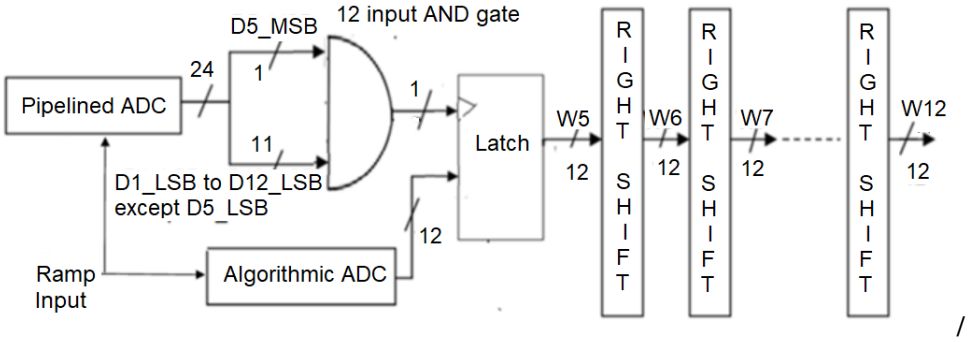

Figure FC5.2: Calculation of weights W_5 – W_{12} of Pipelined ADC

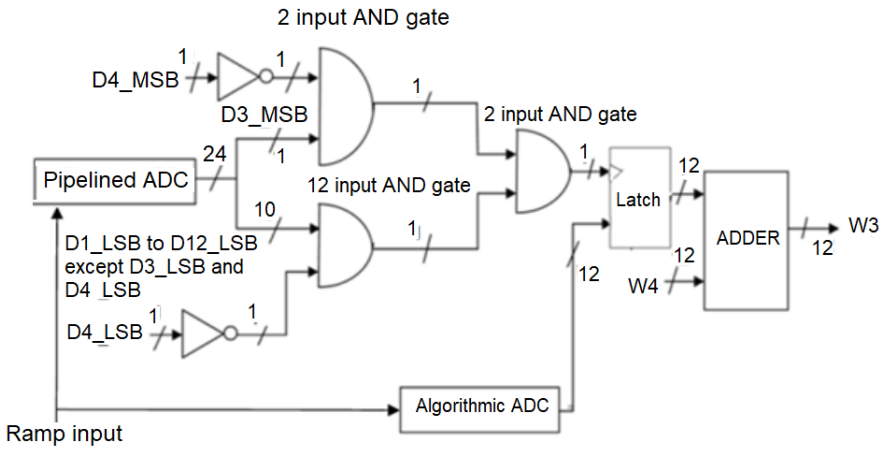

Figure FC5.3: Calculation of weights W_3 of Pipelined ADC

calibrated output of $\frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}$. The representative circuit diagram for the generation of the weight W_3 is as shown in Figure FC5.3. We monitor MSB of the digital codes of the 3^{rd} and 4^{th} stage ($D3\_MSB$ and $D4\_MSB$) to take the digital value 1 and 0 respectively and LSB of the digital codes of the rest of the stages ($D1\_LSB$ to $D12\_LSB$ except $D3\_LSB$ and $D4\_LSB$) to take the value of 1 and $D4\_LSB$ equal to 0. The corresponding digital output bits from the algorithmic ADC are added to W_4 to obtain W_3 [27].

5.1.4 Finding W_2

W_2 can be calculated similar to W_4, by monitoring the digital output of the pipelined ADC for the digital code "0,1,0,0,0,0,0,0,0,0,0,0". The digital equivalent of the analog input voltage, D_{algo4}, is obtained using algorithmic ADC. This corresponds to the pipeline ADC calibrated digital output of $\frac{\beta_2}{\alpha_1.\alpha_2}$ which is equal to W_2 (Eqn 5.1).

5.1.5 Finding W_1

W_1 is calculated similar to W_3 by monitoring the digital output of the pipelined ADC for the digital output code equal to "1,-1,0,0,0,0,0,0,0,0,0,0". This corresponds to a pipeline ADC digital output of $\frac{\beta_1}{\alpha_1}$ - $\frac{\beta_2}{\alpha_1.\alpha_2}$ which is ideally an analog voltage of $\frac{V_{ref}}{4}$. The digital equivalent of the analog input voltage, D_{algo5}, is obtained using algorithmic ADC. The weight, W_1, can be calculated by adding D_{algo5} with D_{algo4} [27].

This completes the calibration phase. During the normal operation of the pipelined ADC, the weights calculated during calibration are used to obtain a calibrated digital code from the pipeline ADC. This is as shown in Eqn 5.2.

Algorithmic ADC, which consists of only one MX2 stage using non-flip-around DAC [7], also needs to be calibrated so that it produces a correct digital equivalent of the analog input voltage and calibration of algorithmic ADC can be done without using precise voltage generator. The advantage of using non-flip-around DAC architecture is that it does not have β and requires calculating only α [7]. The algorithmic ADC can be calibrated using a slow ramp signal (similar to pipeline ADC) instead of a precise voltage generator. The calibrated digital output of the algorithmic ADC, D_{algo}, is obtained using Eqn 5.3 where D_i is the digital output code from the MX2 stage of algorithmic

ADC during i^{th} iteration [7].

$$D_{out} = D_1.\frac{1}{\alpha} + D_2.\frac{1}{\alpha^2} + D_3.\frac{1}{\alpha^3} + ... + D_{11}.\frac{1}{\alpha^{11}} + D_{12}.\frac{1}{\alpha^{12}} \quad \text{(Eqn 5.3)}$$

A ramp signal which is slowly incremented from $\frac{V_{ref}}{4}$ -$\Delta V$to $\frac{V_{ref}}{4}$ + ΔV is given as input to the algorithmic ADC and MSB of the MX2 stage is monitored for a change in the digital code from 0 to +1. At the point when the digital code of the MSB stage changes from 0 to +1, the 12 output digital code of the algorithmic ADC with MSB +1, (D_{MSB1}), Eqn 5.3 and its previous 12 output digital code with MSB 0, (D_{MSB0}), Eqn 5.3 are stored in registers. Relation between D_{MSB1} and D_{MSB0} is given by Eqn 5.4 where 1LSB corresponds to a calibrated digital output of $\frac{1}{\alpha^{12}}$. This equation can be solved using Newton-Raphson method to obtain α. Note that the calibration speed is limited by the speed of the Algorithmic ADC since it uses a non-flip around DAC architecture. The calibrated algorithmic ADC is accurate to 12 bits which otherwise would degrade the performance of the pipeline ADC.

$$D_{MSB1} = D_{MSB0} + 1LSB \quad \text{(Eqn 5.4)}$$

During the calibration of the pipeline ADC, the digital equivalent of the analog input that is obtained from the algorithmic ADC as shown in Eqn 5.3.

5.2 Circuit Implementation

The op-amp that has been used in all MX2 stages works rail-to-rail, and has been designed with a telescopic cascode stage followed by a push-pull common-source stage [22]. The gain achieved by the op-amp is 87 dB with a unity gain frequency of 500 MHz and a phase margin greater than 56^o over all process corners [Appendix. A.1]. The 1.5-bit flash ADC in each stage was designed with Strong-ARM latch comparators [19].

The $\pm \frac{V_{ref}}{4}$ voltage generation for comparators is obtained using a switched-capacitor circuit [20].

The first stage of the pipelined ADC was designed to work as a differential sample and hold amplifier (SHA) during normal operation and reconfigured into a ramp-generating switched-capacitor integrator (SCI) during calibration. This is explained in detail in Chapter 3 (cf. Figure FC3.7). We deduce from the discussion in Sec. 5.1 that the ramp needs to be extremely fine when the input voltages are close to $\frac{V_{ref}}{32}$, $\frac{V_{ref}}{16}$, $\frac{V_{ref}}{4}$. A fine ramp all the way through will also work, but will take an extremely long time to process. The digital output codes required for calibration are produced by the pipeline ADC, when the analog inputs to the ADC are close to $\frac{V_{ref}}{32}$, (measure W_5), $\frac{V_{ref}}{16}$ (measure W_4 and W_3) and $\frac{V_{ref}}{4}$ (measure W_2 and W_1), respectively.

Note that if we adopt the same technique as used in measuring W_5, W_4 and W_2 to measure W_3 and W_1, then it is required to generate extremely fine ramp at voltages close to $\frac{V_{ref}}{8}$ and $\frac{V_{ref}}{2}$ which in turn would take longer for calibration. The ramp signal needs to traverse the voltage range from 0V to 305mV ($\approx \frac{V_{ref}}{4}$) for calibration. The voltage range for the fine ramp depends on the maximum gain error introduced in each stage so that pipeline ADC does not encounter any missing code. For this design, a fine ramp is required for 37.5mV $\pm$ 7.5mV, 75mV $\pm$ 15mV and 300mV $\pm$ 5mV for 5%, 3.75%, 2.5% and 1.25% capacitance mismatch in the 1^{st}, 2^{nd}, 3^{rd} and 4^{th} stages so that the stage gains are 2.05, 2.0375, 2.025 and 2.0125 respectively. Therefore a coarse ramp is generated from 0 to 30mV, 45mV to 60mV and 90mV to 295mV. This is as shown in Figure FC5.4 which also indicates the fine ramp voltages during which the weights W_1 – W_5 are captured.

MX2 stages used in algorithmic ADC works in a ping pong fashion, sampling the input at ϕ_2 (ϕ_1) and producing a residue output at ϕ_1 (ϕ_2) which in turn is sampled by the next MX2 stage at ϕ_1 (ϕ_2). Algorithmic ADC cannot work in a ping pong fashion

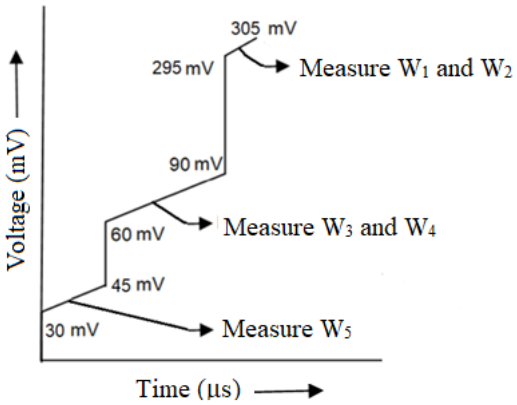

Figure FC5.4: Generation of coarse and fine ramp voltages

as it consists of only one MX2 stage and therefore the residue output of the MX2 stage should be held constant until it is processed by the same MX2 stage again. This can be done by cascading an SHA to the MX2 stage of an algorithmic ADC.

5.3 Simulation Results

The pipelined ADC and algorithmic ADC required for the calibration of pipelined ADC were implemented in a 180-nm CMOS technology of Semiconductor Complex Limited, India (SCL), and were simulated in Cadence-AMS environment with the digital portions coded using Verilog HDL.

The 12 bits pipelined ADC was designed to convert the inputs ranging from 1.2V to -1.2V. The calibration technique was tested using a ramp varying from 1.2V to -1.2V. This was fed to the 4 MSPS pipelined ADC where the capacitance mismatches of the 4 MSB stages are 5%, 3.75%, 2.5% and 1.25%. Note that the proposed calibration scheme can be implemented on pipelined ADCs of much higher speed as the op-amp limits the speed of the pipelined ADC and not the digital hardware used for calibration. The calibrated digital output was input to an ideal DAC to obtain an analog output (for

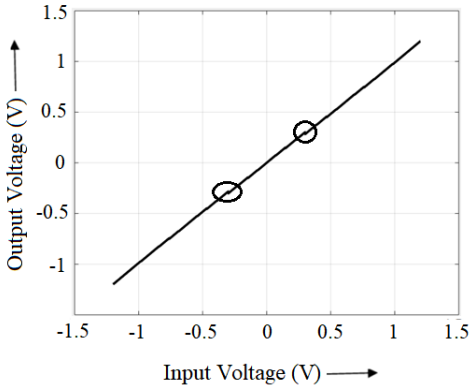

Figure FC5.5: Output transfer characteristics of Pipelined ADC before calibration

ease of visualization). The transfer characteristics of the pipelined ADC before and after calibration are shown in Figure FC5.5 and Figure FC5.6 respectively.

The plot demonstrates that the proposed calibration technique eliminates the discontinuities caused by capacitance mismatch and gain errors and produces a linear output from $-V_{ref}$ to $+V_{ref}$. The non-linearity of the pipelined ADC was measured using the simulation results of the pipelined ADC from Cadence. With, mismatch introduced in the 4 MSB stages, a ramp signal varying from -1.2V to 1.2V was given as input to the ADC and the digital output of the ADC was processed in MATLAB to obtain the INL and DNL of the ADC. The INL plot of the ADC before and after the calibration is as shown in Figure FC5.7. The plot shows that the INL of the ADC after calibration is less than 0.7 LSB which otherwise is close to 30 LSB. Figure FC5.8 shows a plot of SNR and SFDR for full scale analog input sampled at 4MSPS for frequencies varying between 100 KHz and 1.33 M Hz.

Table TC5.1 summarizes the performance of the ADC.

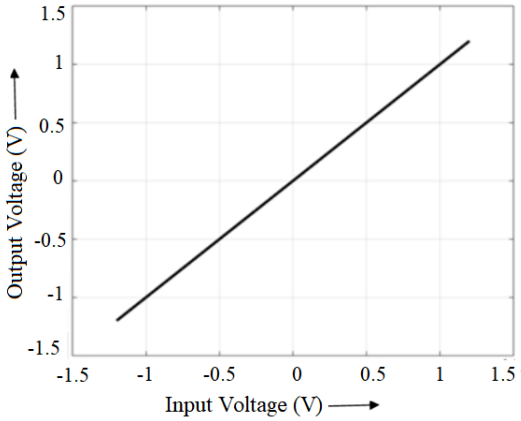

Figure FC5.6: Output transfer characteristics of Pipelined ADC after calibration

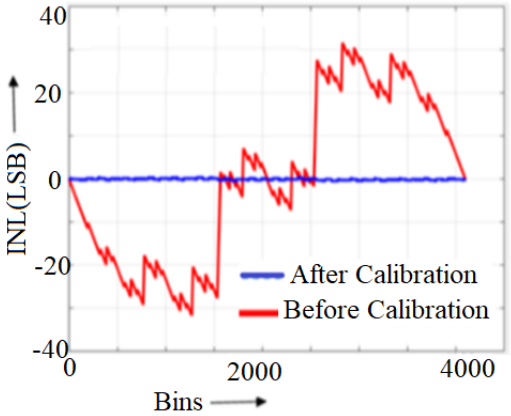

Figure FC5.7: INL plot of the ADC before and after calibration

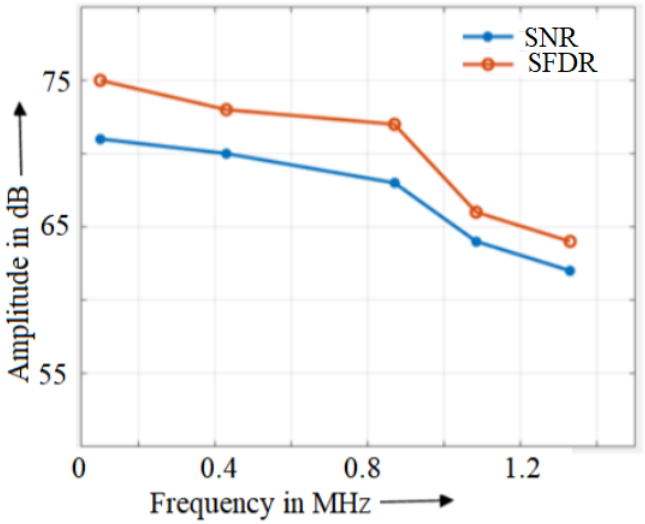

Figure FC5.8: SNR and SFDR plot of a Pipelined ADC

Table TC5.1: Performance table of the pipelined ADC

Resolution	12-bits
Sampling rate	4MHz
Calibration time	250 cycles
SNR	62 dB@1.3MHz, 71 dB @ 100KHz
SFDR	64 dB@1.3MHz, 75 dB @ 100KHz
INL	0.65LSB
DNL	0.3LSB
Input voltage swing	1.2Vp-p
Power dissipation (Without scaling currents in stages)	128mW
Predicted power dissipation by scaling consecutive stages	50mW
Power dissipated by the calibration hardware Algorithmic ADC (SHA +MX2 stage+ digital hardware)	22mW
Estimated area occupied by the digital hardware	127 μm X 130 μm
Estimated area occupied by the analog hardware	650 μm X 750 μm
Process	1.8V, 180nm CMOS
Power supply	1.8V

5.4 Conclusion

A novel foreground calibration technique to calibrate a pipelined ADC is presented in this chapter. The calibration algorithm has been tested on 12 bits, 1.5 bits/stage pipelined ADC. The proposed algorithm requires a slow but accurate ADC which is used to directly extract the digital calibration coefficients. The calibration technique does not require precise generation of any analog voltages. It is seen that the INL and DNL of the calibrated pipeline ADC was reduced to 0.65 LSB and 0.3 LSB respectively. A significant improvement in the SNR and SFDR of the ADC was also observed.

CHAPTER 6

NOVEL DIGITAL CALIBRATION OF ALGORITHMIC ADC AND PIPELINED ADC BY CYCLING MSB STAGES

The chapter proposes a novel digital calibration technique for pipelined ADC and algorithmic ADC which accounts for capacitor mismatches, parasitic capacitances, offset errors and finite open loop gain of the op-amp. A fixed point iterative algorithm is used to determine the weights, with which the digital codes of the pipeline stages of the ADC are multiplied, to obtain calibrated digital output. Unlike the state of the art calibration techniques, the proposed technique uses very less number of samples and cycles for calibration. The calibration algorithm has been implemented on 40 MSPS, 1.5 bits/stage; 12 bits pipeline ADC where 4 MSB stages are calibrated. For a 5% mismatch in capacitances in the 4 MSB stages of the pipelined ADC and open loop gain of the op-amp (A) equal to 52dB, the calibration technique improved SNDR by more than 23dB and SFDR by around 29dB. The technique can also be slightly modified to calibrate algorithmic ADCs. For a 5% mismatch in capacitances of the algorithmic ADC and A = 52dB, the proposed calibration technique improved SNDR by 22dB and SFDR by around 29dB.

6.1 Proposed Algorithm to Calibrate Pipeline ADC

Each pipeline stage of the ADC consists of a multiply-by-2 amplifier (MX2) and comparators. Capacitor mismatch, parasitic capacitances and finite op-amp gain in the MX2 lead to discontinuities and the change in the slope in the output transfer characteristics of the ADC (Cf. FC3.1). These errors can be corrected by calculating the calibration coefficients for each stage. The calibration coefficients are calculated using fixed point iterative algorithm by cycling the pipeline stages. The fixed point iterative algorithm is used to solve N non-linear equations with N unknowns with minimal computation [28]. Let F(X) be a set of N non-linear equations represented as:

$$\begin{aligned} \mathrm{f}_1(x_1, x_2, ...x_n) &= 0 \\ \mathrm{f}_2(x_1, x_2, ...x_n) &= 0 \\ &\vdots \\ \mathrm{f}_n(x_1, x_2, ...x_n) &= 0 \end{aligned}$$

Now, F(X) can be transformed into another set of equations of the form X = G(X), which can be elaborated as follows.

$$\begin{aligned} \mathrm{x}_1 &= \mathrm{g}_1(x_1, x_2, ...x_n) \\ \mathrm{x}_2 &= \mathrm{g}_2(x_1, x_2, ...x_n) \\ &\vdots \\ \mathrm{x}_n &= \mathrm{g}_n(x_1, x_2, ...x_n) \end{aligned}$$

With the conditions of contraction mapping theorem satisfied, we calculate the fixed points, X, starting with an initial guess of X(0). Then, subsequent iterations are computed as follows,

$$X^{K+1} = G(X^K), K = 0, 1, 2... \qquad \text{(Eqn 6.1)}$$

The equations F(X) and hence G(X) depends on capacitor mismatches, parasitic capacitances and finite open loop gain of the op-amp. The set of equations G(X) sat-

isfies contraction mapping theorem for a finite open loop gain of 52 dB and capacitor mismatches from $\pm 1\%$ to $\pm 10\%$.

An empirical understanding of the MX2 stage is required to use fixed point iteration to obtain the calibration coefficients. A detailed working of the MX2 stage along with the equations are given in Chapter 4 (cf. FC4.1).

The goal of the calibration technique is to measure the actual magnitudes of α and β to obtain a calibrated digital output. The operations for calculating α and β of each stage are performed on the digital output codes and their values are stored in a fixed point format. To demonstrate the algorithm, we consider a 1.5 bits/stage, 12 stages ADC, where the calibration coefficients of 4 MSB stages are to be calculated assuming ideal backend stages (stages 5 to 12). Assuming stages 5-12 are ideal, the digital equivalent of the input voltage to the 5^{th} stage is calculated as

$$D_{in,5} = D_5.2^{-1} + D_6.2^{-2} + D_7.2^{-3} + D_8.2^{-4} + D_9.2^{-5} + D_{10}.2^{-6} + D_{11}.2^{-7} + D_{12}.2^{-8} \qquad \text{(Eqn 6.2)}$$

The block diagram of pipeline ADC is as shown in Figure FC6.1. V_{in}, refers to the input analog signal to be converted by the ADC. V_{cal} refers to the calibration input signal set to $\frac{V_{ref}}{4}$. During the normal operation of the ADC, the switches are connected as follows. Switch S_1 connects the external signal to be converted by the ADC, V_{in}, to the input of stage-1. Switches S_5 and S_2 connect the residue output of the stage-1 to the input of the stage-2. Switches S_6 and S_3 connect the residue output of the stage-2 to the input of the stage-3. Switches S_7 and S_4 connect the residue output of the stage-3 to the input of the stage-4. Switches S_8 and S_9 connect the residue output of the stage-4 to the input of the stage-5.

The switches S_1-S_9 are used to cycle the stages to calculate the calibration coeffi-

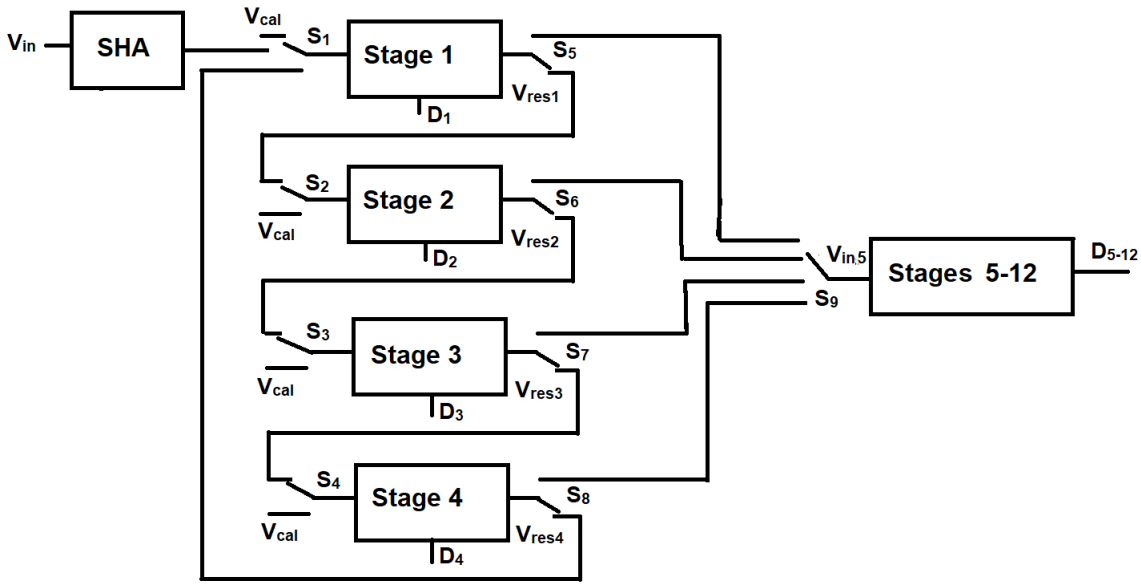

Figure FC6.1: Block diagram of Pipeline ADC

cients.

Fixed point iterative algorithm is used to calculate the actual values of calibration coefficients (α, β) of the 4 MSB stages. The proposed algorithm takes 5-10 iterations to converge to the actual values of α and β for capacitance mismatches upto $\pm 10\%$ in the stages to be calibrated when simulated with finite op-amp gain of 52dB.

6.1.1 Calculating α_1 and β_1

During the calculation of the calibration coefficients in the first stage, the switches are connected in the following manner. Switch S_1 connects the calibration signal, ($V_{cal} = \frac{V_{ref}}{4}$) to the input of stage-1. Switches S_5 and S_2 connect the residue output of the stage-1 to the input of the stage-2. Switches S_6 and S_3 connect the residue output of the stage-2 to the input of the stage-3. Switches S_7 and S_4 connect the residue output of the stage-3 to the input of the stage-4. Switches S_8 and S_9 connect the residue output of the stage-4 to the input of the stage-5.

The block diagram of the pipeline ADC shown in Figure FC6.1 reduces to Figure FC6.2.

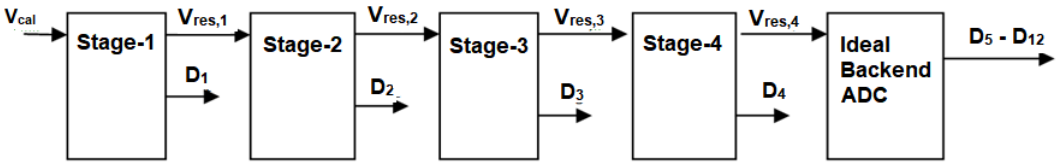

Figure FC6.2: Reduced block diagram of Pipeline ADC

With the input to the ADC, V_{cal} ($=\frac{V_{ref}}{4}$) (as shown in Figure FC6.2) and the digital output code of the 1^{st} stage, D_1, forced to code 0, the residue voltage of 1^{st} stage is calculated. The residue equation for the first stage is given by $V_{res,1}(D_1 = 0)$ as shown,

$$V_{res,1}(D_1 = 0) = \alpha_1 . \frac{V_{ref}}{4} \qquad \text{(Eqn 6.3)}$$

For $V_{in} = \frac{V_{ref}}{4}$ and D_1 forced to code 1, the residue voltage of the 1^{st} stage, $V_{res,1}(D_1{=}1)$, is,

$$V_{res,1}(D_1 = 1) = \alpha_1 . \frac{V_{ref}}{4} - \beta_1 . V_{ref} \qquad \text{(Eqn 6.4)}$$

The digital equivalent of the analog residue voltages, $V_{res,1}$, are obtained by dividing Eqn 6.3 and Eqn 6.4 by V_{ref}.

$$D_{res,1\_0} = \frac{\alpha_1}{4} \qquad \text{(Eqn 6.5)}$$

$$D_{res,1\_1} = \frac{\alpha_1}{4} - \beta_1 \qquad \text{(Eqn 6.6)}$$

It is required to obtain the values of $D_{res,1\_0}$ and $D_{res,1\_1}$ in order to calculate α_1 and β_1. We calculate the approximate digital equivalent of the residue voltages, $D_{res,1\_0}$ and

$D_{res,1\_1}$, as shown in Eqn 6.7 and Eqn 6.8, using the 11 digital codes from stage 2 to stage 12 of the pipeline ADC with an initial estimate of $\alpha_2 = \alpha_3 = \alpha_4 = 2$, $\beta_2 = \beta_3 = \beta_4 = 1$. During the calculation of D_{res,i_j}, let $D_{i\_j\_k}$ be the digital codes of the pipeline stages where i represents the stage being calibrated, j represents the digital output code that is forced to stage i, k represents the stage to which the digital code correspond to. The digital equivalent of the residue voltages is not the correct estimate of analog residue voltages as α_n and β_n (n = 2,3,4) used to calculate $D_{res,1}$ are not accurate.

$$D_{res,1\_0} = D_{1\_0\_2}.\frac{\beta_2}{\alpha_2} + D_{1\_0\_3}.\frac{\beta_3}{\alpha_2.\alpha_3} + D_{1\_0\_4}.\frac{\beta_4}{\alpha_2.\alpha_3.\alpha_4} + \frac{D_{in,5\_0\_1}}{\alpha_2.\alpha_3.\alpha_4} \qquad \text{(Eqn 6.7)}$$

$$D_{res,1\_1} = D_{1\_1\_2}.\frac{\beta_2}{\alpha_2} + D_{1\_1\_3}.\frac{\beta_3}{\alpha_2.\alpha_3} + D_{1\_1\_4}.\frac{\beta_4}{\alpha_2.\alpha_3.\alpha_4} + \frac{D_{in,5\_1\_1}}{\alpha_2.\alpha_3.\alpha_4} \qquad \text{(Eqn 6.8)}$$

where $D_{in,5\_0\_1}$ represents the digital equivalent of input to the 5^{th} stage when 1^{st} stage is being calibrated with its digital code forced to 0, $D_{in,5\_1\_1}$ represents the digital equivalent of input to the 5^{th} stage when 1^{st} stage is being calibrated with its digital code forced to 1.

By substituting Eqn 6.7 and Eqn 6.8 in Eqn 6.5 and Eqn 6.6, the updated values of α_1 and β_1 can be expressed in the required form as shown,

$$\alpha_{1\_u} = 4 * D_{res,1\_0} = g_1(\alpha_2, \alpha_3, \alpha_4, \beta_2, \beta_3, \beta_4) \qquad \text{(Eqn 6.9)}$$

$$\beta_{1\_u} = D_{res,1\_0} - D_{res,1\_1} = g_2(\alpha_2, \alpha_3, \alpha_4, \beta_2, \beta_3, \beta_4) \qquad \text{(Eqn 6.10)}$$

The updated values of α_1 and β_1 are calculated from Eqn 6.9 and Eqn 6.10. Let us call the value of α_1 and β_1 as α_{1_u} and β_{1_u} which act as intermediate values to calculate

the final values of α_1 and β_1.

6.1.2 Calculating α_4 and β_4

Now, the pipeline stages 1-4 are right shifted in a circular manner such that the 4^{th} stage now becomes the MSB pipeline stage, the 1^{st} stage becomes the 2^{nd} pipeline stage, 2^{nd} stage becomes the 3^{rd} pipeline stage and 3^{rd} stage becomes the 4^{th} pipeline stage. This is as shown in Figure FC6.3.

Switch S_4 connects the calibration signal, $(V_{cal} = \frac{V_{ref}}{4})$ to the input of stage-4. Switches S_8 and S_1 connect the residue output of the stage-4 to the input of the stage-1. Switches S_5 and S_2 connect the residue output of the stage-1 to the input of the stage-2. Switches S_6 and S_3 connect the residue output of the stage-2 to the input of the stage-3. Switches S_7 and S_9 connect the residue output of the stage-3 to the input of the stage-5. Now, with $\alpha_1 = \alpha_{1\_u}$, $\alpha_2 = \alpha_3 = 2$, $\beta_1 = \beta_{1\_u}$ $\beta_2 = \beta_3 = 1$, α_4 and β_4 are calculated. With input voltage $V_{cal} = \frac{V_{ref}}{4}$ and forcing $D_4 = 0$ and 1 respectively, we calculate the residue outputs of stage 4, $V_{res4}(D_4 = 0)$ and $V_{res4}(D_4 = 1)$.

$$V_{res,4}(D_4 = 0) = \alpha_4 . \frac{V_{ref}}{4} \qquad \text{(Eqn 6.11)}$$

$$V_{res,4}(D_4 = 1) = \alpha_4 . \frac{V_{ref}}{4} - \beta_4 . V_{ref} \qquad \text{(Eqn 6.12)}$$

We calculate the approximate digital equivalent of the residue voltages as,

$$D_{res,4\_0} = \frac{\alpha_4}{4} \qquad \text{(Eqn 6.13)}$$

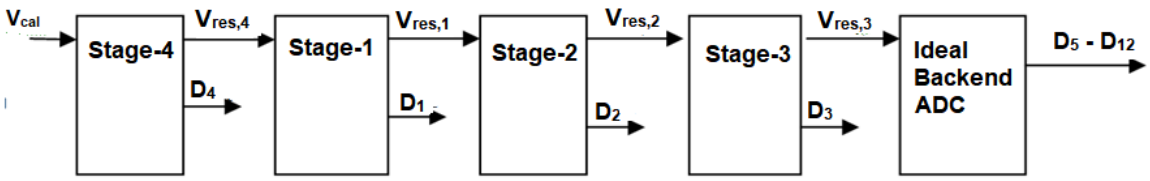

Figure FC6.3: After shifting the pipeline stages of ADC shown in FC6.2

$$D_{res,4\_1} = \frac{\alpha_4}{4} - \beta_4 \qquad \text{(Eqn 6.14)}$$

The approximate digital equivalents of the residue voltages $D_{res,4\_0}$ and $D_{res,4\_1}$ are obtained from 11 digital codes D_1, D_2, D_3 and D_5 to D_{12} of the pipeline ADC with $\alpha_1 = \alpha_{1\_u}$, $\alpha_2 = \alpha_3 = 2$, $\beta_1 = \beta_{1\_u}$, $\beta_2 = \beta_3 = 1$ as shown in Eqn 6.15 and Eqn 6.16. Note that $D_{res,4\_0}$ and $D_{res,4\_1}$ are not accurate since α's, β's used in the calculation of $D_{res,4}$ are only approximate values.

$$D_{res,4\_0} = D_{4\_0\_1}.\frac{\beta_1}{\alpha_1} + D_{4\_0\_2}.\frac{\beta_2}{\alpha_1.\alpha_2} + D_{4\_0\_3}.\frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3} + \frac{D_{in,5\_0\_4}}{\alpha_1.\alpha_2.\alpha_3} \qquad \text{(Eqn 6.15)}$$

$$D_{res,4\_1} = D_{4\_1\_1}.\frac{\beta_1}{\alpha_1} + D_{4\_1\_2}.\frac{\beta_2}{\alpha_1.\alpha_2} + D_{4\_1\_3}.\frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3} + \frac{D_{in,5\_1\_4}}{\alpha_1.\alpha_2.\alpha_3} \qquad \text{(Eqn 6.16)}$$

where $D_{in,5\_0\_4}$ represents the digital equivalent of input to the 5^{th} stage when 4^{th} stage is being calibrated with its digital code forced to 0, $D_{in,5\_1\_4}$ represents the digital equivalent of input to the 5^{th} stage when 4^{th} stage is being calibrated with its digital code forced to 1.

By substituting Eqn 6.15 and Eqn 6.16 in Eqn 6.13 and Eqn 6.14, the updated values of α_4 and β_4 can be expressed in the required form as shown,

$$\alpha_{4\_u} = 4 * D_{res,4\_0} = g_3(\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3) \qquad \text{(Eqn 6.17)}$$

$$\beta_{4\_u} = D_{res,4\_0} - D_{res,4\_1} = g_4(\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3) \quad \text{(Eqn 6.18)}$$

The intermediate values of α_{4_u} and β_{4_u} are calculated similar to α_{1_u}, β_{1_u}.

6.1.3 Calculating α_3 and β_3

Now, the pipeline stages are again right shifted in a circular manner such that the 3^{rd} stage now becomes the MSB stage, the 4^{th} stage becomes the 2^{nd} pipeline stage, 1^{st} stage becomes the 3^{rd} pipeline stage and 2^{nd} stage becomes the 4^{th} pipeline stage.

Switch S_3 connects the calibration signal, $(V_{cal} = \frac{V_{ref}}{4})$ to the input of stage-3. Switches S_7 and S_4 connect the residue output of the stage-3 to the input of the stage-4. Switches S_8 and S_1 connect the residue output of the stage-4 to the input of the stage-1. Switches S_5 and S_2 connect the residue output of the stage-1 to the input of the stage-2. Switches S_6 and S_9 connect the residue output of the stage-2 to the input of the stage-5. This is as shown in Figure FC6.4.

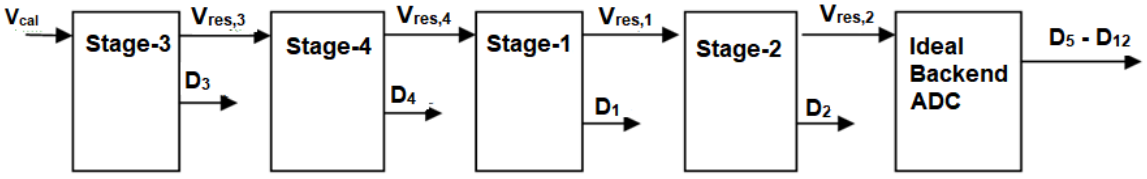

Figure FC6.4: After shifting the pipeline stages of ADC shown in FC6.3

At this point, with input $V_{cal} = \frac{V_{ref}}{4}$ and D_3 forced to 0 and 1 respectively, we calculate the residue output voltages of stage 3, $V_{res3}(D3 = 0)$ and $V_{res3}(D3 = 1)$. The intermediate value of α_{3_u} and β_{3_u} are calculated similar to α_{4_u} and β_{4_u} by calculating approximate digital equivalents $D_{res,3\_0}$ and $D_{res,3\_1}$ as shown in Eqn 6.19 and

Eqn 6.20.

$$D_{res,3\_0} = D_{3\_0\_4}.\frac{\beta_4}{\alpha_4} + D_{3\_0\_1}.\frac{\beta_1}{\alpha_1.\alpha_4} + D_{3\_0\_2}.\frac{\beta_2}{\alpha_4.\alpha_1.\alpha_2} + \frac{D_{in,5\_0\_3}}{\alpha_4.\alpha_1.\alpha_2} \qquad \text{(Eqn 6.19)}$$

$$D_{res,3\_1} = D_{3\_1\_4}.\frac{\beta_4}{\alpha_4} + D_{3\_1\_1}.\frac{\beta_1}{\alpha_1.\alpha_4} + D_{3\_1\_2}.\frac{\beta_2}{\alpha_4.\alpha_1.\alpha_2} + \frac{D_{in,5\_1\_3}}{\alpha_4.\alpha_1.\alpha_2} \qquad \text{(Eqn 6.20)}$$

Where $\alpha_1 = \alpha_{1\_u}$, $\alpha_4 = \alpha_{4\_u}$, and $\alpha_2 = 2$, $\beta_1 = \beta_{1\_u}$, $\beta_4 = \beta_{4\_u}$ and $\beta_2 = 2$. With this, we obtain the equations $g_5(\alpha_1, \alpha_2, \alpha_4, \beta_2, \beta_4, \beta_1)$, $g_6(\alpha_1, \alpha_2, \alpha_4, \beta_2, \beta_4, \beta_1)$ to calculate α_{3_u} and β_{3_u}.

6.1.4 Calculating α_2 and β_2

Now, the pipeline stages are again right shifted in a circular motion such that the 2^{nd} stage now becomes the MSB pipeline stage, the 3^{rd} stage becomes the 2^{nd} pipeline stage, 4^{th} stage becomes the 3^{rd} pipeline stage and 1^{st} stage becomes the 4^{th} pipeline stage. The intermediate updated values of α_{2_u} and β_{2_u} are calculated with $\alpha_1 = \alpha_{1\_u}$, $\alpha_4 = \alpha_{4\_u}$, $\alpha_3 = \alpha_{3\_u}$, $\beta_1 = \beta_{1\_u}$, $\beta_4 = \beta_{4\_u}$ and $\beta_3 = \beta_{3\_u}$ and from equations $g_7(\alpha_1, \alpha_3, \alpha_4, \beta_3, \beta_4, \beta_1)$, $g_8(\alpha_1, \alpha_3, \alpha_4, \beta_3, \beta_4, \beta_1)$. This completes one cycle of calculating digital calibration coefficients.

Eqn 6.1 is used to update the values of α's and β's iteratively. With the Digital Code sets ($D_{i\_j\_k}$) already known, the digital portion of calibration algorithm is executed until all α's and β's reach the fixed points. This is as shown in FC6.5, for capacitor mismatch of -10% in stage 1, -5% in stage 2, +10% in stage 3 and +5% in stage 4 with finite open loop gain of 52dB in all stages.

This marks the end of the calibration phase. During normal operation of the ADC, the values of α_i and β_i (for i=1,2,3,4) thus obtained are used to calculate the calibrated digital output. For an analog input V_{in}, the corrected (calibrated) digital output, D_{out}, is

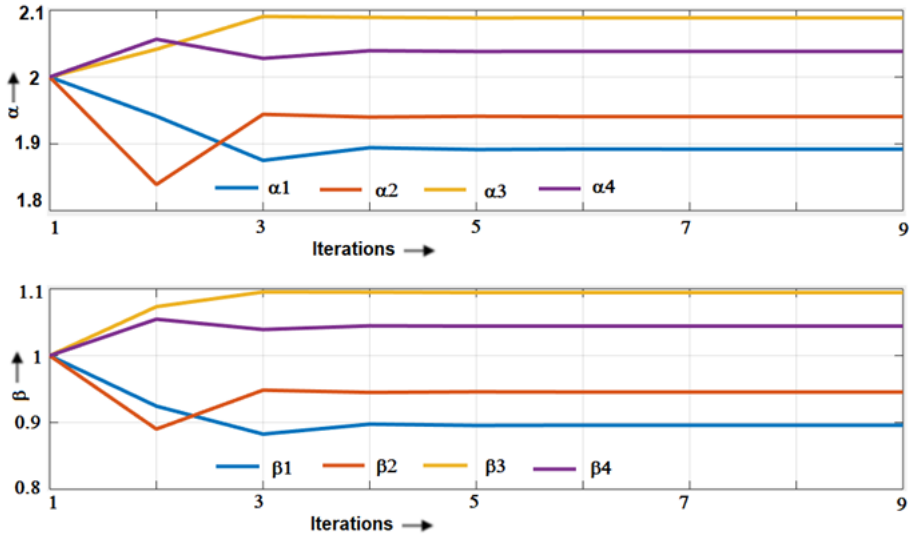

Figure FC6.5: Convergence of α's and β's in the pipelined ADC

given by Eqn 6.21.

$$D_{out} = D_1.\frac{\beta_1}{\alpha_1} + D_2.\frac{\beta_2}{\alpha_1.\alpha_2} + D_3.\frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3} + D_4.\frac{\beta_4}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} + D_5.\frac{2^{-1}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}$$
$$+D_6.\frac{2^{-2}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} + D_7.\frac{2^{-3}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} + D_8.\frac{2^{-4}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} + D_9.\frac{2^{-5}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} +$$
$$D_{10}.\frac{2^{-6}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} + D_{11}.\frac{2^{-7}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4} + D_{12}.\frac{2^{-8}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}$$

(Eqn 6.21)

where D_1 to D_{12} are the digital output codes of stages 1 to 12 respectively.

6.2 Proposed Algorithm to Calibrate Algorithmic ADC

Algorithmic ADC consists of only one MX2 stage and hence it is required to calculate α and β of one stage. 12 bits algorithmic ADC takes a new input every 12 cycles. During the rest of the cycles, the output of the pipeline stage is fed back to its input. Therefore algorithmic ADC takes 12 cycles to convert the sampled analog input to its

digital equivalent. Neglecting offset voltage, the residue output of the MX2 stage during the n^{th} iteration, $V_{res,n}$, at the end of the amplification phase can be written as [11],

$$V_{res,n} = \frac{(C_2 + C_1).V_{in}(n) - V_{ref}.C_2.D_n}{C_1 + \frac{C_2+C_p+C_1}{A}} \quad \text{(Eqn 6.22)}$$

C_1, C_2, C_p and A are the feedback, sampling, parasitic capacitors and open loop gain of the op-amp, respectively, $V_{in}(n)$ is the analog input during the n^{th} iteration (note that $V_{in}(n) = V_{res,n-1}$ where $V_{res,n-1}$ is the analog residue output voltage of the previous iteration), V_{ref} is the reference voltage and D_n is the digital output code of the MX2 stage during n^{th} iteration:

$$D_n = \begin{cases} +1\ ;\ if\ V_{res,n-1} > +\frac{V_{ref}}{4}, \\ -1\ ;if\ V_{res,n-1} < -\frac{V_{ref}}{4}, \\ 0\ ;Otherwise. \end{cases}$$

Eqn 6.22 can be re-written in terms of α and β as shown in Eqn 4.3, where

$$\alpha = \frac{C_1 + C_2}{C_1 + \frac{(C_2+C_1+C_p)}{A}}$$
$$\beta = \frac{C_2}{C_1 + \frac{(C_2+C_1+C_p)}{A}} \quad \text{(Eqn 6.23)}$$

The residue output of the MX2 stage during the n^{th} iteration, $V_{res,n}$, can be re-written as,

$$V_{res,n} = \alpha.V_{in,n} - V_{ref}.\beta.D_n \equiv \alpha.V_{res,n-1} - V_{ref}.\beta.D_n \quad \text{(Eqn 6.24)}$$

The calibrated digital output of the ADC at the end of 12 clock cycles is given by,

$$D_{algo} = D_1.\frac{\beta}{\alpha} + D_2.\frac{\beta}{\alpha^2} + D_3.\frac{\beta}{\alpha^3} + D_4.\frac{\beta}{\alpha^4} + D_5.\frac{\beta}{\alpha^5} + D_6.\frac{\beta}{\alpha^6} + D_7.\frac{\beta}{\alpha^7} + D_8.\frac{\beta}{\alpha^8} + D_9.\frac{\beta}{\alpha^9} + D_{10}.\frac{\beta}{\alpha^{10}} + D_{11}.\frac{\beta}{\alpha^{11}} + D_{12}.\frac{\beta}{\alpha^{12}} \quad \text{(Eqn 6.25)}$$

6.2.1 Calculating α and β

With the input to the ADC, V_{in}, forced to $\frac{V_{ref}}{4}$ and the digital output code of the 1^{st} iteration, D_1, forced to code 0, residue output of the MX2 stage is given by $V_{res}(D_1 = 0)$ as shown,

$$V_{res}(D_1 = 0) = \alpha.\frac{V_{ref}}{4} \quad \text{(Eqn 6.26)}$$

For V_{in} = $\frac{V_{ref}}{4}$ and D_1 = 1, the residue voltage $V_{res}(D_1 = 1)$, is

$$V_{res}(D_1 = 1) = \alpha.\frac{V_{ref}}{4} - \beta.V_{ref} \quad \text{(Eqn 6.27)}$$

The digital equivalent of the analog residue voltage, D_{res} is obtained by dividing Eqn 6.26, Eqn 6.27, by V_{ref}.

$$D_{res0} = \frac{\alpha}{4} \quad \text{(Eqn 6.28)}$$

$$D_{res1} = \frac{\alpha}{4} - \beta \quad \text{(Eqn 6.29)}$$

It is required to obtain the values of D_{res1} and D_{res0} in order to calculate α and β. We calculate the approximate digital equivalent of the residue voltages, D_{res1} and D_{res0}, as shown in Eqn 6.30 using the 11 digital codes (D_2–D_{12}) of the algorithmic ADC with an

initial estimate of α = 2 and β = 1 .

$$D_{res} = D_2.\frac{\beta}{\alpha} + D_3.\frac{\beta}{\alpha^2} + D_4.\frac{\beta}{\alpha^3} + D_5.\frac{\beta}{\alpha^4} + D_6.\frac{\beta}{\alpha^5} + D_7.\frac{\beta}{\alpha^6} + D_8.\frac{\beta}{\alpha^7} + D_9.\frac{\beta}{\alpha^8} + D_{10}.\frac{\beta}{\alpha^9} + D_{11}.\frac{\beta}{\alpha^{10}} + D_{12}.\frac{\beta}{\alpha^{11}} \quad \text{(Eqn 6.30)}$$

By substituting D_{res1} and D_{res0} obtained from Eqn 6.30 in Eqn 6.28 and Eqn 6.29, α and β can be expressed in the required form as shown,

$$\alpha_u = 4 * (D_{res0}) = g_1(\alpha, \beta) \quad \text{(Eqn 6.31)}$$

$$\beta_u = D_{res0} - D_{res1} = g_2(\alpha, \beta) \quad \text{(Eqn 6.32)}$$

Let α_u and β_u be the updated values of α and β which are calculated using Eqn 6.31 and Eqn 6.32. Note that α_u and β_u are not the final values of α and β.

This completes one cycle of calculating digital calibration coefficients. With the digital code sets already known, the digital portion of calibration algorithm is executed to update the calibration coefficients until α and β converge to its fixed point. This marks the end of the calibration phase.

During normal operation of the ADC, the values of α and β thus obtained are used to calculate the calibrated digital output as in Eqn 6.25.

6.3 Circuit implementation

The op-amp that has been used in SHA stages works rail-to-rail, and has been designed with a telescopic cascode stage followed by a push-pull common-source stage [22]. The gain achieved by the op-amp is 87 dB with a unity gain frequency of 500 MHz and a phase margin greater than 56^o over all process corners [Appendix A.1]. The

op-amp that has been used in all MX2 stages works rail-to-rail, and has been designed for a gain of 52dB with a unity gain frequency of 700 MHz and a phase margin greater than 61^o [Appendix A.2].

The 1.5-bit flash ADC in each stage was designed with Strong-ARM latch comparators . The $\pm\frac{V_{ref}}{4}$ voltage generation for comparators is obtained using a switched-capacitor circuit [20]. The differential sample and hold circuit was built using a switched capacitor circuit [19]. The reference voltage, $\frac{V_{ref}}{4}$ required to generate the datasets for calibration can be obtained as in [7].

6.3.1 Pipelined ADC

There are 2 modes of operation.

1. Normal mode
2. Calibration mode

There are few static control signals such as: CAL, CAL1, CAL2, CAL3, CAL4 and FRC. All the control signals are a combination of these signals.

CAL: Enables the calibration mode.

CALn: Enables n^{th} stage for calibration where n = 1,2,3,4.

- CAL1 enables the following connections. Switch S_1 connects the calibration signal, ($V_{cal} = \frac{V_{ref}}{4}$) to the input of stage1. Switches S_5 and S_2 connects the residue output of the stage1 to the input of the stage2. Switches S_6 and S_3 connects the residue output of the stage2 to the input of the stage3. Switches S_7 and S_4 connects the residue output of the stage3 to the input of the stage4. Switches S_8 and S_9 connects the residue output of the stage4 to the input of the stage5.
- CAL4 enables the following connections. Switch S4 connects the calibration sig-

nal, ($V_{cal} = \frac{V_{ref}}{4}$) to the input of stage4. Switches S8 and S1 connects the residue output of the stage4 to the input of the stage1. Switches S5 and S2 connects the residue output of the stage1 to the input of the stage2. Switches S6 and S3 connects the residue output of the stage2 to the input of the stage3. Switches S7 and S9 connects the residue output of the stage3 to the input of the stage5.

- CAL3 enables the following connections. Switch S3 connects the calibration signal, ($V_{cal} = \frac{V_{ref}}{4}$) to the input of stage3. Switches S7 and S4 connects the residue output of the stage3 to the input of the stage4. Switches S8 and S1 connects the residue output of the stage4 to the input of the stage1. Switches S5 and S2 connects the residue output of the stage1 to the input of the stage2. Switches S6 and S9 connects the residue output of the stage2 to the input of the stage5.

- CAL2 enables the following connections. Switch S2 connects the calibration signal, ($V_{cal} = \frac{V_{ref}}{4}$) to the input of stage2. Switches S6 and S3 connects the residue output of the stage2 to the input of the stage3. Switches S7 and S4 connects the residue output of the stage3 to the input of the stage4. Switches S8 and S1 connects the residue output of the stage4 to the input of the stage1. Switches S5 and S9 connects the residue output of the stage1 to the input of the stage5.

FRC: Signal for forcing the two comparators to the desired region. During calibration mode, signal FRC when set high forces a digital code of +1 at the output of the pipeline stage if CAL signal of the particular stage is set to 1. When set to low, forces a digital code of 0 by overriding the output of the comparator. Table TC6.1 shows the configuration of FRC signal during calibration and normal mode of operation.

Table TC6.2 shows the input configuration of control signals during calibration mode in the same order. All the digital backend codes thus obtained during calibration are stored in registers to be reused during iterations until actual values of α and β are obtained.

Table TC6.1: Configuration of FRC signal

CALn	FRC	Remarks
1	0	Force digital output code of pipeline stage 'n' to 0 with its input = V_{cal}
1	1	Force digital output code of pipeline stage 'n' to 1 with its input = V_{cal}
0	X	Normal mode of operation

Table TC6.2: Input Configuration of control signals during calibration mode

Mode of Operation	CAL	CAL1	CAL2	CAL3	CAL4	FRC	Remarks
Updating $\alpha 1$	1	1	0	0	0	0	$\alpha 1 = \frac{D_{res1,0}}{4}$
Updating $\beta 1$	1	1	0	0	0	1	$\beta 1 = D_{res1,0} - D_{res1,1}$
Updating $\alpha 4$	1	0	0	0	1	0	$\alpha 4 = \frac{D_{res4,0}}{4}$
Updating $\beta 4$	1	0	0	0	1	1	$\beta 4 = D_{res4,0} - D_{res4,1}$
Updating $\alpha 3$	1	0	0	1	0	0	$\alpha 3 = \frac{D_{res3,0}}{4}$
Updating $\beta 3$	1	0	0	1	0	1	$\beta 3 = D_{res3,0} - D_{res3,1}$
Updating $\alpha 2$	1	0	1	0	0	0	$\alpha 2 = \frac{D_{res2,0}}{4}$
Updating $\beta 2$	1	0	1	0	0	1	$\beta 2 = D_{res2,0} - D_{res2,1}$

After one cycle of updating the calibration coefficients, iterative algorithm continues with its second cycle of updating the calibration coefficients and so on until all the calibration coefficients converge to their actual values. During this phase, the signals *CALn* and FRC are set low and signal *CAL* is set high to indicate the process of calibration coefficient calculation is not complete yet. The signal CAL is set low after all the calibration coefficients converge to their actual values. This completes the calibration phase.

6.3.2 Algorithmic ADC

Algorithmic ADC requires calibration of one stage. There are 2 modes of operation.

1. Normal mode
2. Calibration mode

There are 2 static control signals: CAL and FRC.

CAL enables the calibration mode when set high. The input of the Algorithmic ADC is connected to ($V_{cal} = \frac{V_{ref}}{4}$) and FRC is set low to force digital code of 0 at the output. In the subsequent 11 clock cycles the residue output of the algorithmic ADC is connected to its input. The 11 output digital codes thus obtained are stored in registers. Now, the input of the Algorithmic ADC is connected to ($V_{cal} = \frac{V_{ref}}{4}$) and FRC is set high to force digital code of +1 at the output. In the subsequent 11 clock cycles the residue output of the algorithmic ADC is connected to its input. The 11 output digital codes thus obtained are stored in registers. These output digital codes are used to obtain the actual values of α and β using the iterative algorithm.

6.4 Simulation Results

The pipelined ADC and algorithmic ADC along with the calibration scheme were implemented in a 180-nm CMOS technology of Semiconductor Complex Limited, India (SCL), and were simulated in Cadence-AMS environment with the digital portion coded in verilog. The op-amp has also been simulated across the process corners and results are promising. SHA built using this op-amp shows a minimal overshot with the final value of the output settling with an error of 60μV for a full scale input when operated at 40 MHz [Appendix A]. The pipelined ADC works for the inputs ranging from -1.2V to 1.2V.

To validate the calibration algorithm, a ramp signal varying from -1.2V to 1.2V is input to the designed 40MSPS pipelined ADC where the capacitance mismatches varies from 1% up to 10%. The digital output thus obtained is calibrated with the proposed calibration algorithm and fed to an ideal DAC to obtain the analog output for ease of visualization. This is as shown in Figure FC6.6. It can be seen that the linearity of the output transfer characteristics of the ADC improves after calibration. Note that the

proposed calibration scheme can be implemented on pipelined ADCs of much higher speed as the op-amp limits the speed of the pipelined ADC and not the digital hardware used for calibration.

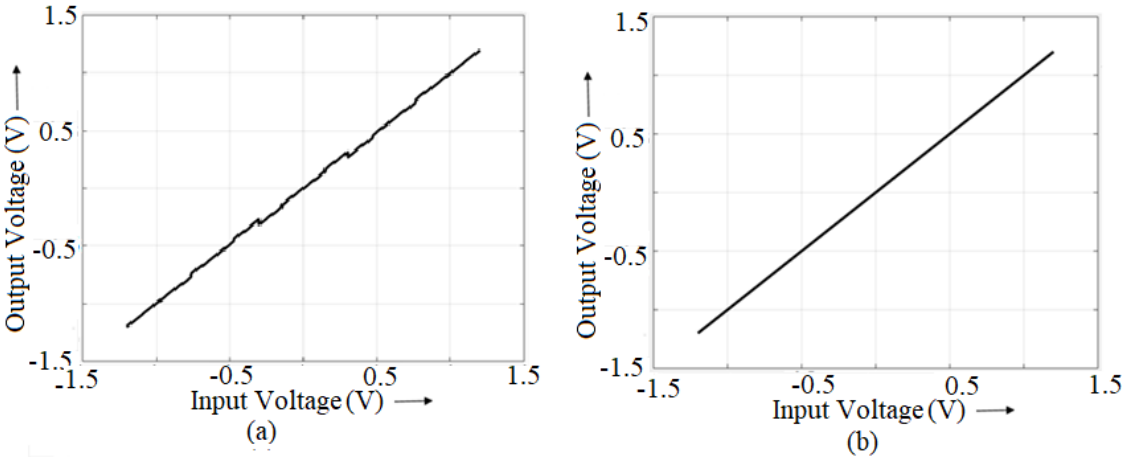

Figure FC6.6: Output transfer characteristics of pipeline ADC (a) Before calibration (b) After calibration

To validate the calibration algorithm on algorithmic ADC, an input ramp signal varying from -1.2V to 1.2V is input to the designed 4MSPS algorithmic ADC where the capacitance mismatches varies up to 5%. The digital output thus obtained is calibrated with the proposed calibration algorithm and fed to an ideal DAC to obtain the analog output for ease of visualization. It can be seen that the linearity of the output transfer characteristics of the ADC improves after calibration. This is as shown in Figure FC6.7.

Figure FC6.8a shows the output spectrum of a 12.7MHz full scale analog sine input which is sampled at 40 MSPS (N=1024) by the pipelined ADC, before and after calibration. It is seen that SFDR improves from 36dB before calibration to 65 dB after calibration. Figure FC6.8b shows the output spectrum of a 1.5MHz full scale analog sine input which is sampled at 4 MSPS(N=2048) by the pipelined ADC, before and after calibration. It is seen that SFDR improves from 37dB before calibration to 66 dB after calibration. The Integral non linearity (INL) of the pipelined ADC and algorith-

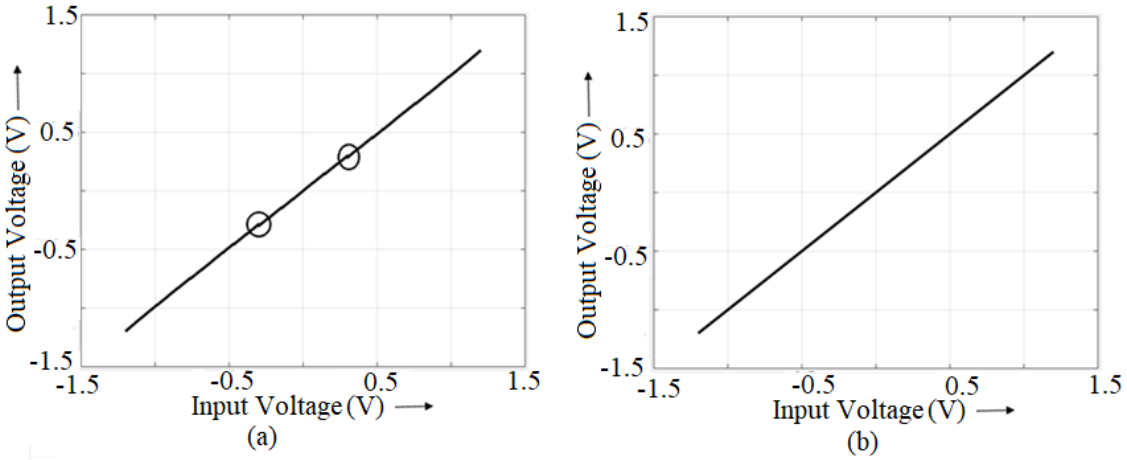

Figure FC6.7: Output transfer characteristics of algorithmic ADC (a) Before calibration (b) After calibration

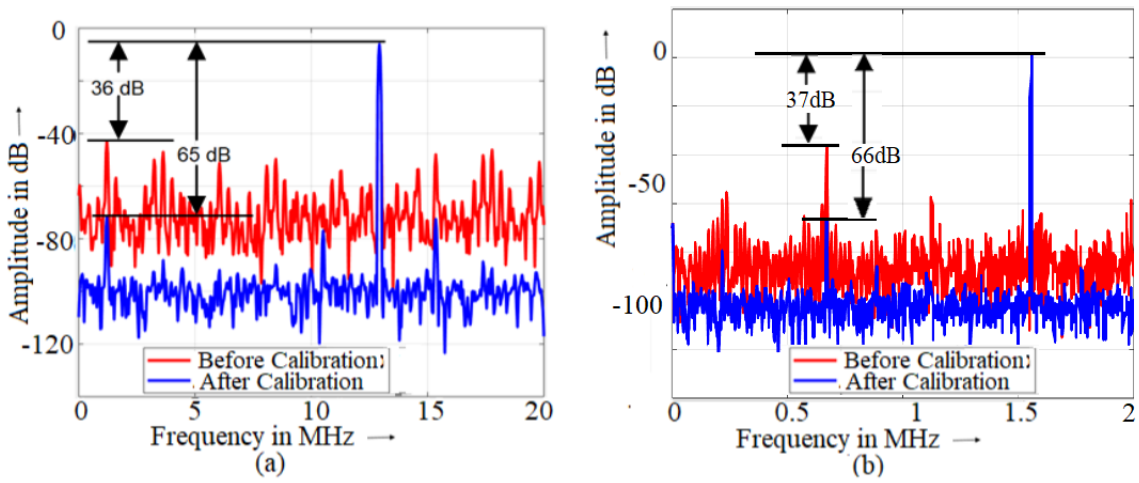

Figure FC6.8: Output spectrum of (a) Pipeline ADC and (b)Algorithmic ADC before and after calibration

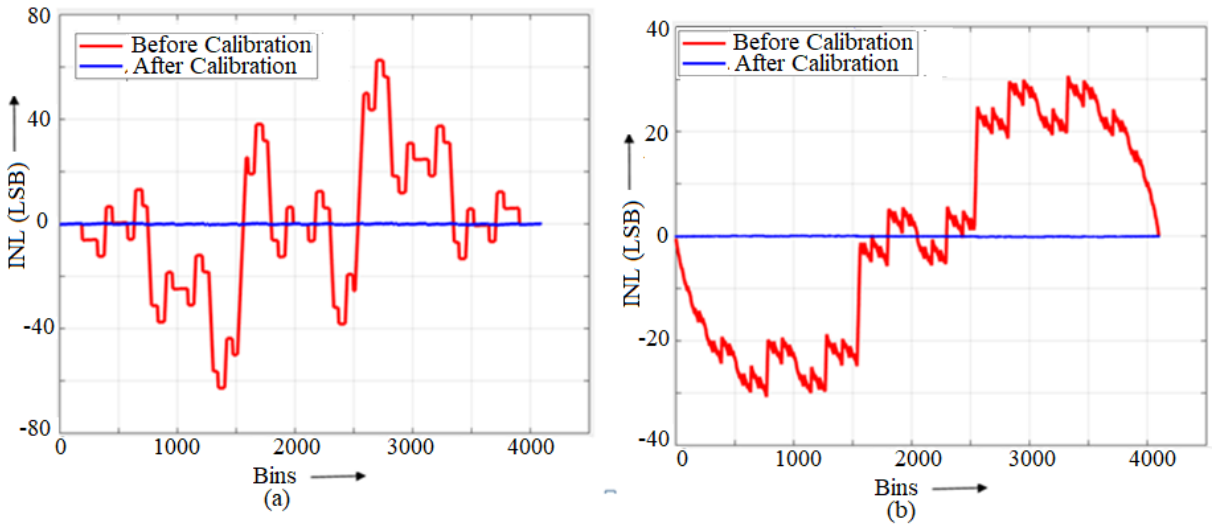

Figure FC6.9: INL of (a) Pipeline ADC and (b)Algorithmic ADC before and after calibration

mic ADC before and after calibration are as shown in Figure FC6.9a and Figure FC6.9b. The INL of the pipeline and algorithmic ADC after calibration reduces from 60 LSB to 0.7 LSB and 30 LSB to 0.65 LSB respectively for up to 5% variation in the capacitance mismatch.

The proposed method obtains α and β values to a higher accuracy. It is necessary that these values are stored and manipulated to this higher accuracy (e.g. for a 12-bit ADC where four MSBs are calibrated, the α and β values must be stored and manipulated to 16 bits).

Figure FC6.10 shows an improvement in SNDR for a full scale analog input before and after calibration of the pipelined and algorithmic ADC. Table TC6.3 gives the performance of the pipelined ADC and algorithmic ADC respectively.

6.5 Conclusions

A novel deterministic digital calibration technique is proposed to correct both capacitance mismatches as well as gain errors, in 1.5-bits/stage pipelined as well as algo-

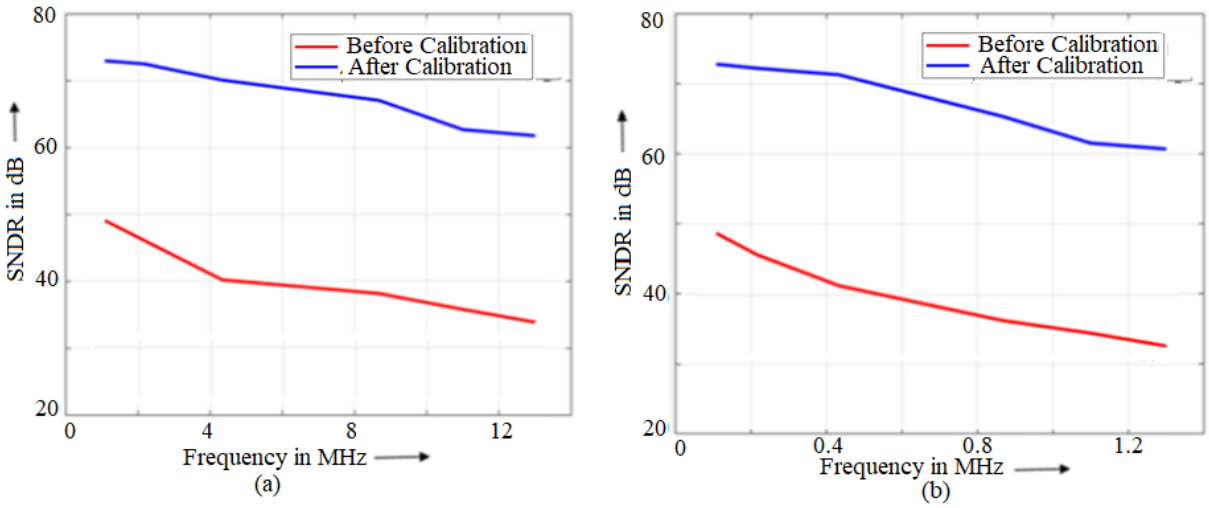

Figure FC6.10: SNDR of (a) Pipeline ADC and (b)Algorithmic ADC before and after calibration

Table TC6.3: Performance table of the 1.5 bits/stage pipelined ADC and algorithmic ADC

	Pipeline	Algorithmic
Resolution	12-bits	12-bits
Sampling rate	40MHz	4MHz
SNR	62 dB@13MHz, 73 dB @ 1MHz	60dB @1.5MHz 73 dB @ 100KHz
SFDR	65 dB@13MHz	66dB @1.5MHz
INL	0.7 LSB	0.65 LSB
DNL	0.29 LSB	0.32 LSB
Input voltage swing	1.2Vp-p	1.2Vp-p
Power dissipation (Without scaling currents in stages)	50mW	12mW
Predicted power dissipation by scaling consecutive stages	32mW	-
Process	1.8V, 180nm CMOS	1.8V, 180nm CMOS

rithmic ADCs. To illustrate the technique, a 12bits, 1.5-bits/stage pipelined ADC, and a 1.5-bit stage algorithmic ADC were designed in a 180nm technology. With close to 5% mismatch in capacitance values, INL and DNL of the calibrated pipelined ADC were reduced to 0.7 LSB and 0.29 LSB, respectively, and the INL and DNL of the calibrated algorithmic ADC were reduced to 0.65 LSB and 0.32 LSB. A significant improvement

in the SNDR, ENOB and SFDR after calibration in both pipelined as well as algorithmic ADC was also observed.

CHAPTER 7

BACKGROUND DETERMINISTIC DIGITAL CALIBRATION OF 1.5 BITS/STAGE PIPELINED ADC BY SUBSTITUTING THE PIPELINE STAGES

This chapter describes two novel background calibration techniques to digitally calibrate pipeline ADCs. The proposed deterministic calibration techniques account for capacitor mismatches in the pipeline stage, finite op-amp gain, comparator offsets and charge injection. The techniques have been tested on a 40 MSPS 12 bit pipelined ADC where the 4 most significant stages are calibrated. The idea behind the calibration technique is that each pipeline stage of the ADC that undergoes calibration is disconnected from the pipeline ADC and substituted by a pre-calibrated extra pipeline stage until the calibration of the disconnected pipeline stage completes. This ensures that the conversion process by pipeline ADC is not hampered. The proposed technique does not require any DSP or microprocessor for calibration and requires fewer samples and cycles for calibration. For up to 6.25% mismatch in capacitances and open loop op-amp gain of 52dB in the 4 MSB stages, both the calibration techniques improved SNR by more than 25dB and SFDR by around 33dB at close to Nyquist input frequency.

7.1 Proposed Algorithm

The proposed calibration algorithms are demonstrated with the help of a 1.5 bits/stage, 12 stage pipelined ADC where 4 MSB stages are calibrated. Each pipeline stage consists of MX2 stage and 2 comparators to obtain the output digital code. The analog residue output of each pipeline stage is fed as input to the succeeding pipeline stage. The output digital codes of the pipeline stages are connected to digital block which is a combination of an array of D flip-flops and calibration block to obtain calibrated digital output. The circuit and working of MX2 is explained briefly in Chapter 4. The goal of the calibration technique is to compute α and β of the 4 MSB stages in the background and use them during the normal operation of the ADC to obtain calibrated output.

7.1.1 Calibration Technique 1

During the calibration phase, calibration coefficients of the 4 MSB stages are measured by treating each pipeline stage as one stage algorithmic ADC. Calibration algorithm requires an Extra Sample and Hold Amplifier [ESHA], extra pipeline stage and an additional digital block which is made up of an array of D flip-flops and calibration coefficient calculation block to calculate α and β. The procedure to calculate the calibration coefficients for pipeline stage1 has been demonstrated in this section. This procedure holds good for calculating calibration coefficients of all the stages to be calibrated. The pipeline stage1 is disconnected from the pipeline ADC for calibration and replaced with an extra pipeline stage so that the actual working of the ADC is not hindered. A pictorial representation of pipeline stage1, being swapped with the extra pipeline stage during calibration is as shown in Figure FC7.1. The pipeline stage1 is connected to ESHA and D flip-flop array from additional digital block as shown in Figure FC7.1 to form one stage algorithmic ADC. The outputs of D flip-flop array from

additional digital block are in-turn connected to the calibration coefficient calculation block to calculate α and β. The calibration coefficients of the algorithmic ADC are calculated using Newton Raphson technique as explained in Chapter 4. This completes the calibration phase of one pipeline stage. The same procedure is repeated for the other pipeline stages that require calibration.

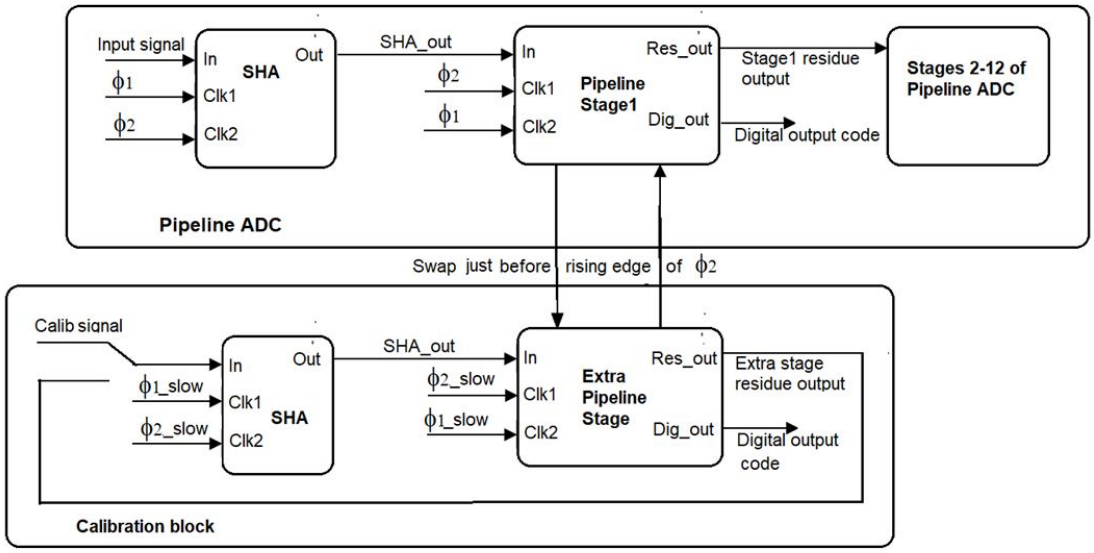

Figure FC7.1: Replacement of pipeline stages during calibration (Technique 1)

Following are to be noted during calibration.

- If the pipeline stage to be calibrated, samples the new input when $\phi 2$ is high and produces the residue output when $\phi 1$ is high, the replacement of the pipeline stages for calibration should happen just before $\phi 2$ goes high. Otherwise, the residue output from the previous stage, required to obtain the output digital codes and hence the residue output of the current stage is lost. Similarly, the pipeline stage that samples the new input when $\phi 1$ is high and produces the residue output when $\phi 2$ is high, has to be replaced just before $\phi 1$ goes high. Thus, the time at which the pipeline stages are swapped during calibration is very important.
- The calibration coefficients do not account for the finite op-amp gain of the ESHA.

Linearity of the ESHA can be improved by using a larger capacitance in the circuit which compensates for the finite op-amp gain of ESHA. With this larger capacitance, the speed with which the calibration is performed reduces. Therefore, the calibration of the pipeline stages is performed at a lower speed when compared to the speed of the ADC.

- During the calibration of any pipeline stage, the extra pipeline stage replaces the pipeline stage under calibration in the pipeline ADC. The clock signal of the extra pipeline stage has to be switched from lower speed to the higher speed. As a precautionary measure, the high speed clock is connected to the extra pipeline stage one cycle before it replaces the pipeline stage under calibration to avoid any settling problems in the pipeline ADC. Once the calibration of the pipeline stage is completed, again, high speed clock is connected to the pipeline stage one cycle before it replaces the extra pipeline stage after calibration.

Once, the digital calibration coefficients α and β are measured for all the stages undergoing calibration, they are used to get a calibrated digital output code, D_{out}, of the pipelined ADC as shown in Eqn 7.1. This is as explained in detail in Chapter 4.

$$D_{out} = \frac{\beta_1}{\alpha_1}.D_1 + \frac{\beta_2}{\alpha_1.\alpha_2}.D_2 + \frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}.D_3 + \frac{\beta_4}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}.D_4 + \frac{D_5}{2^1.\alpha_1.\alpha_2.\alpha_3.\alpha_4} + ... + \frac{D_{12}}{2^8.\alpha_1.\alpha_2.\alpha_3.\alpha_4} \quad \text{(Eqn 7.1)}$$

7.1.2 Calibration Technique 2

Although the finite gain error of the ESHA can be reduced by choosing a higher order capacitor, it cannot be completely eliminated. This problem can be overcome by using 2 extra pipeline stages, $pipe\_ex1$ and $pipe\_ex2$. With this, the calibration coefficients of the 2 pipeline stages of the pipeline ADC can be calculated at a time. Two

stages of the pipeline ADC that are required to be calibrated, are substituted with the already calibrated extra stages, $pipe\_ex1$ and $pipe\_ex2$ in the pipeline ADC. The 2 stages of the pipeline ADC that are disconnected from the pipeline ADC can be calibrated together treating them as 2 stage cyclic ADC. Note that the 2 MSB stages of the pipeline ADC cannot be substituted at once as it leads to the loss of the intermediate data. If pipeline stage1 and pipeline stage2 are substituted by extra stages at the same time, then the data that is being transferred from stage1 to stage2 during the time of substitution is lost. Also $pipe\_ex2$ will not have any input data to be converted. If 2 MSB stages of the pipeline ADC has to undergo calibration, the MSB stage is substituted after the residue output of the present data is calculated and before a new data is sampled onto the MSB stage similar to one stage algorithmic ADC. Since the pipelined ADC works in a ping pong fashion, the 2^{nd} MSB stage is substituted half a clock cycle later, after the residue output of the present data is calculated and before the new data is sampled onto the 2^{nd} MSB stage. This is as shown in Figure FC7.2.

Calibration technique similar to the one used in algorithmic ADC cannot be used to calibrate a cyclic ADC since there are 4 unknowns (α_1 and β_1 of the MSB stage and α_2 and β_2 of the 2^{nd} MSB stage) instead of 2. Fixed point iterative algorithm can be used to calculate the calibration coefficients of 2 stages at a time. This technique does not require swapping of clock signals and hence the calibration algorithm works at the speed of the pipeline ADC. The proposed algorithm can be used to calibrate two stage cyclic ADC as well.

Let $pipe\_1$ and $pipe\_2$ be 2 stages to be calibrated. Let α_1, β_1, α_2 and β_2 be the calibration coefficients of stages $pipe\_1$ and $pipe\_2$ respectively. During the initial iteration, let, $\alpha_1 = \alpha_2 = 2$ and $\beta_1 = \beta_2 = 1$. By forcing the input of the $pipe\_1$ to $\frac{V_{ref}}{4}$ and digital output code D_1 of the $pipe\_1$ to 0, the residue output of the $pipe\_1$ stage is

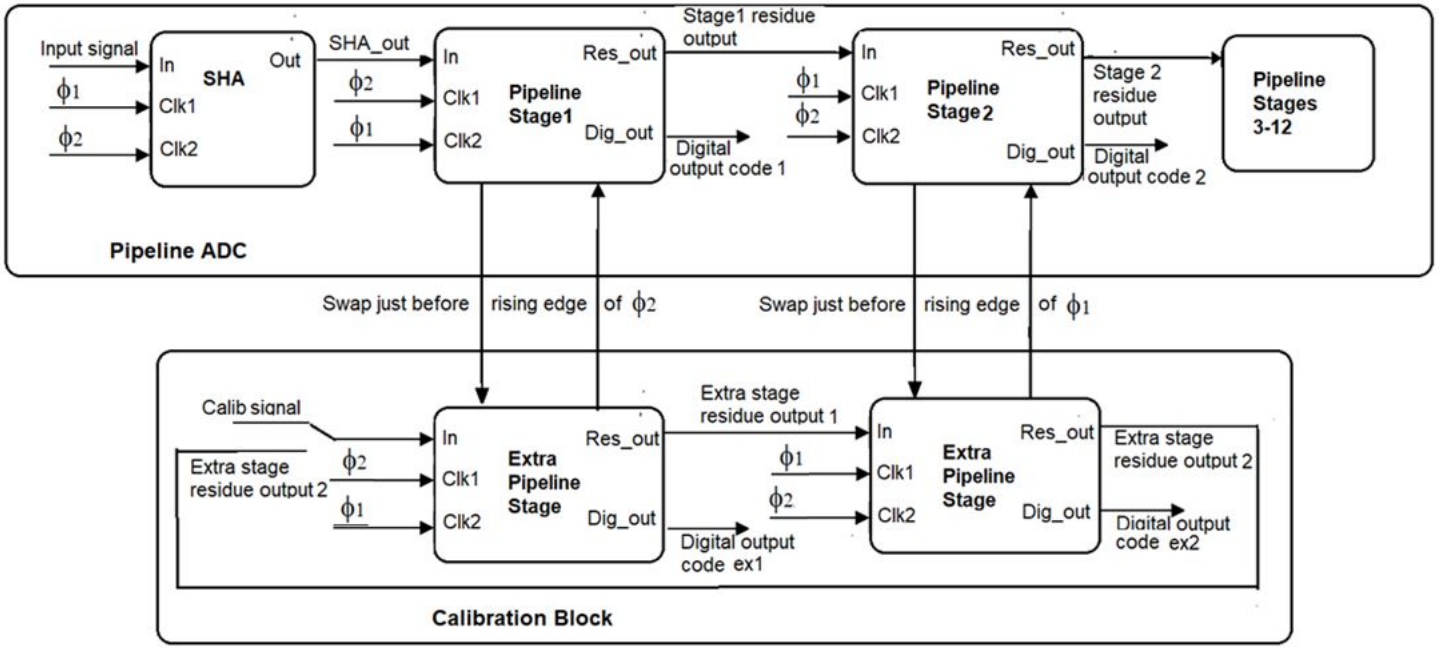

Figure FC7.2: Replacement of pipeline stages during calibration (Technique 2)

given by Eqn 7.2. (Refer Chapter 4)

$$V_{res,1\_0} = \alpha_1 \frac{V_{ref}}{4} \quad \text{(Eqn 7.2)}$$

The approximate digital equivalent of $V_{res,1\_0}$, D_{out1_0}, is obtained by dividing Eqn 7.2 by V_{ref}.

$$D_{out1\_0} = \frac{\alpha_1}{4} \quad \text{(Eqn 7.3)}$$

D_{out1_0} is obtained with the help of 11 digital codes from D_2 to D_{12} of the cyclic ADC after 6 cycles (Most significant code forced to 0) with α_1 = α_2 =2 and β_1 = β_2 = 1 as shown in Eqn 7.4. Note that D_{out1_0} is not accurate since α's and β's used in the calculation of D_{out1_0} are only approximate values.

$$\begin{aligned} D_{out1\_0} = {} & \frac{\beta_2}{\alpha_2}.D_2 + \frac{\beta_1}{\alpha_1.\alpha_2}.D_3 + \frac{\beta_2}{\alpha_1.\alpha_2.\alpha_1}.D_4 + \frac{\beta_1}{(\alpha_1.\alpha_2)^2}.D_5 + \frac{\beta_2}{\alpha_2.(\alpha_1.\alpha_2)^2}.D_6 \\ & + \frac{\beta_1}{(\alpha_1.\alpha_2)^3}.D_7 + \frac{\beta_2}{\alpha_2.(\alpha_1.\alpha_2)^3}.D_8 + \frac{\beta_1}{(\alpha_1.\alpha_2)^4}.D_9 + \frac{\beta_2}{\alpha_2.(\alpha_1.\alpha_2)^4}.D_{10} \\ & + \frac{\beta_1}{(\alpha_1.\alpha_2)^5}.D_{11} + \frac{\beta_2}{\alpha_2.(\alpha_1.\alpha_2)^5}.D_{12} \end{aligned} \quad \text{(Eqn 7.4)}$$

Where D_2 is the output digital code obtained from $pipe\_2$ during the first cycle, D_3 and D_4 are the output digital codes obtained from $pipe\_1$ and $pipe\_2$ respectively during the second cycle and so on. The new value of α_1 is calculated by substituting $D_{out,1\_0}$, which is calculated from Eqn 7.4 in Eqn 7.3. Let us call this value of α_1 as α_{1_u} as shown in Eqn 7.5. Note that α_{1_u} is not the final value of α_1. The value of α_{1_u} is inaccurate as $D_{out,1\_0}$ is calculated with initial estimates of α_1 = α_2 = 2 and β_1 = β_2 = 1 which is not accurate. However, this acts as an intermediate value to calculate the

final value of α_1.

$$\alpha_{1\_u} = 4.D_{out1\_0} \quad \text{(Eqn 7.5)}$$

By forcing the input of the $pipe\_1$ to $\frac{V_{ref}}{4}$ and digital output code, D_1, of the $pipe\_1$ to 1, the residue output of the $pipe\_1$ stage is given by Eqn 7.6.

$$V_{res,1\_1} = \alpha_1.\frac{V_{ref}}{4} - V_{ref}.\beta_1 \quad \text{(Eqn 7.6)}$$

Digitizing Eqn 7.6 gives

$$D_{out,1\_1} = \frac{\alpha_1}{4} - \beta_1 \quad \text{(Eqn 7.7)}$$

$D_{out,1\_1}$ is obtained using digital codes D_2 to D_{12} of the cyclic ADC with α_1 = α_2 =2 and β_1 = β_2 = 1. (cf. Eqn 7.4). The new value of β_1, β_{1_u}, is calculated as shown in Eqn 7.8. Similar to α_{1_u}, β_{1_u} is not the final value of β_1 but serves as an intermediate value to calculate the final value of β_1.

$$\beta_{1\_u} = D_{out,1\_0} - D_{out,1\_1} \quad \text{(Eqn 7.8)}$$

α_{1_u} and β_{1_u} are now used to calculate α_{2_u} and β_{2_u}. α_{2_u} and β_{2_u} are calculated similar to α_{1_u} and β_{1_u} with an initial estimate of α_1 = α_{1_u}, β_1 = β_{1_u}, α_2 =2 and β_2 = 1 by forcing the input of $pipe\_2$ to $\frac{V_{ref}}{4}$ and MSB output digital code of $pipe\_2$ to 0 and 1 respectively. The digital representation of the residue output voltage of stage 2 is

given by D_{out2} . This is as shown in Eqn 7.9.

$$D_{out2} = \frac{\beta_1}{\alpha_1}.D_2 + \frac{\beta_2}{\alpha_1.\alpha_2}.D_3 + \frac{\beta_1}{\alpha_2.\alpha_1.\alpha_2}.D_4 + \frac{\beta_2}{(\alpha_1.\alpha_2)^2}.D_5 + \frac{\beta_1}{\alpha_1.(\alpha_1.\alpha_2)^2}.D_6 + \frac{\beta_2}{(\alpha_1.\alpha_2)^3}.D_7 + \frac{\beta_1}{\alpha_1.(\alpha_1.\alpha_2)^3}.D_8 + \frac{\beta_2}{(\alpha_1.\alpha_2)^4}.D_9 + \frac{\beta_1}{\alpha_1.(\alpha_1.\alpha_2)^4}.D_{10} + \frac{\beta_2}{(\alpha_1.\alpha_2)^5}.D_{11} + \frac{\beta_1}{\alpha_1.(\alpha_1.\alpha_2)^5}.D_{12} \quad \text{(Eqn 7.9)}$$

Where D_2 is the output digital code obtained from $pipe\_1$ during the first cycle, D_3 and D_4 are the output digital codes obtained from $pipe\_2$ and $pipe\_1$ respectively during the second cycle and so on. The approximate digital equivalent of the residue voltage of $pipe\_2$ with its most significant output digital code forced to 0 and 1 respectively (D_{out2_0} and D_{out2_1}), are calculated from Eqn 7.9. Using D_{out2_0} and D_{out2_1}, the updated values of α_2 and β_2 (α_{2_u} and β_{2_u}) are obtained similar to α_{1_u} and β_{1_u}.

This completes one iteration of calculations. The next iteration of calculating α_1 and β_1 starts with an estimate of $\alpha_1 = \alpha_{1\_u}$, $\beta_1 = \beta_{1\_u}$, $\alpha_2 = \alpha_{2\_u}$ and $\beta_2 = \beta_{2\_u}$. This process is repeated until both α's and β's converge to its actual values. The above technique can be used to calibrate all the required stages of the pipeline ADC. Once, α and β of all the stages are calculated, they can be used to calculate the calibrated digital code of the pipelined ADC.

The rate of convergence of α_1 and α_2 are as shown in Figure FC7.3 for capacitance mismatch of +7.5% in the first stage and -5% in the second stage. It is seen that α converges to its final value in about 5 cycles. The comparison table for calibrating pipeline ADC by substituting one stage at a time and two stages at a time is tabulated in Table TC7.1.

During data conversion by the pipelined ADC, initially, the output digital codes are error prone until all the MSB stages are calibrated. Once the calibration cycle starts

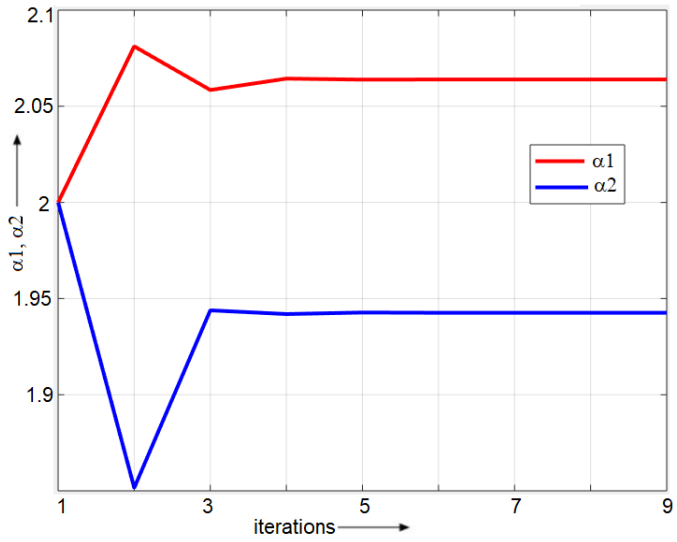

Figure FC7.3: Rate of convergence of α's with $\alpha_1 = 2.075$ and $\alpha_2 = 1.95$

Table TC7.1: Comparison table for calibrating pipeline ADC by substituting one stage at a time and two stages at a time

Calibration Technique 1	**Calibration Technique 2**
Extra hardware required for calibration includes one pipeline stage and one SHA along with additional digital block	Extra hardware required for calibration includes 2 pipeline stages along with additional digital block
Only one Pipeline stage is calibrated at a time	Two pipeline stages are calibrated at a time
Calibration clock speed is decreased to overcome the errors introduced by the SHA	Calibration takes place at the same speed as that of the pipelined ADC
Requires extra signals to swap the clock signals during calibration	Swapping of clock signals are not required during calibration

calibrating pipeline stages one by one, the linearity of the ADC gradually improves.

7.2 Circuit Implementation

7.2.1 Pipelined ADC

Op-amp forms a major building block of all the MX2 stages and SHA. The op-amp used in the SHA works rail to rail and is designed with a telescopic cascade stage followed by push pull common source stage [22]. The op-amp works with a unity gain

frequency of 500MHz and gain and phase margin of 87dB and 56^o over all process corners. The differential sample and hold amplifier used in the pipelined ADC is designed as in [21] with a bootstrap switch (Appendix A.3) at the input. The differential MX2 stage is as shown in Chapter 4. The op-amp that has been used in all MX2 stages works rail-to-rail, and has been designed with a 5T differential amplifier with NMOS input pair in the first stage followed by a common-source stage. The gain achieved by the op-amp is 52dB with a unity gain frequency of 700 MHz and a phase margin greater than 59^o (Appendix A.2). The 1.5-bit flash ADC (Appendix A.5) in each stage was designed with Strong-ARM latch comparators (Appendix A.4) [19]. The $\pm\frac{V_{ref}}{4}$ voltage generation for comparators is obtained using the switched capacitor circuit [20].

The representative digital block diagram to calculate calibrated digital codes is as shown in Figure FC7.4. α_n and β_n represent the calibration coefficients of the n^{th} stage respectively. The inputs to all the multiplexers are represented in fixed point format. The select lines to the multiplexer are the output digital code from the pipeline stage that can take a value of 1, 0 or -1. These are represented using 2 digital bits, An_M and An_L, where An_M represents the MSB of the n^{th} pipeline stage and An_L represents the LSB. The output of the first multiplexer is $-\frac{\beta_1}{\alpha_1}$ if the most significant output digital code of the pipeline stage is '-1' represented by the digital bits $A1\_M = 0$ and $A1\_L = 0$, 0 if $A1\_M = 0$ and $A1\_L = 1$ and $\frac{\beta_1}{\alpha_1}$ if $A1\_M = 1$ and $A1\_L = 0$ respectively. Assuming stages 5-12 are ideal, the inputs to the 2^{nd}, 3^{rd}, 4^{th} and 5^{th} multiplexers are 0 and $\pm\frac{\beta_2}{\alpha_1.\alpha_2}$, $\pm\frac{\beta_3}{\alpha_1.\alpha_2.\alpha_3}$ $\pm\frac{\beta_4}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}$ and $\pm\frac{2^{-1}}{\alpha_1.\alpha_2.\alpha_3.\alpha_4}$ respectively. The outputs of all the multiplexers are then added to get the 12 bit calibrated digital output of the pipelined ADC.

7.2.2 Calibrating one stage at a time

Let the pipeline stages required to be calibrated in a 12 stage ADC are extra pipeline stage (*stage_ex*), pipeline stage1 (*stage_1*), pipeline stage2 (*stage_2*), pipeline stage3

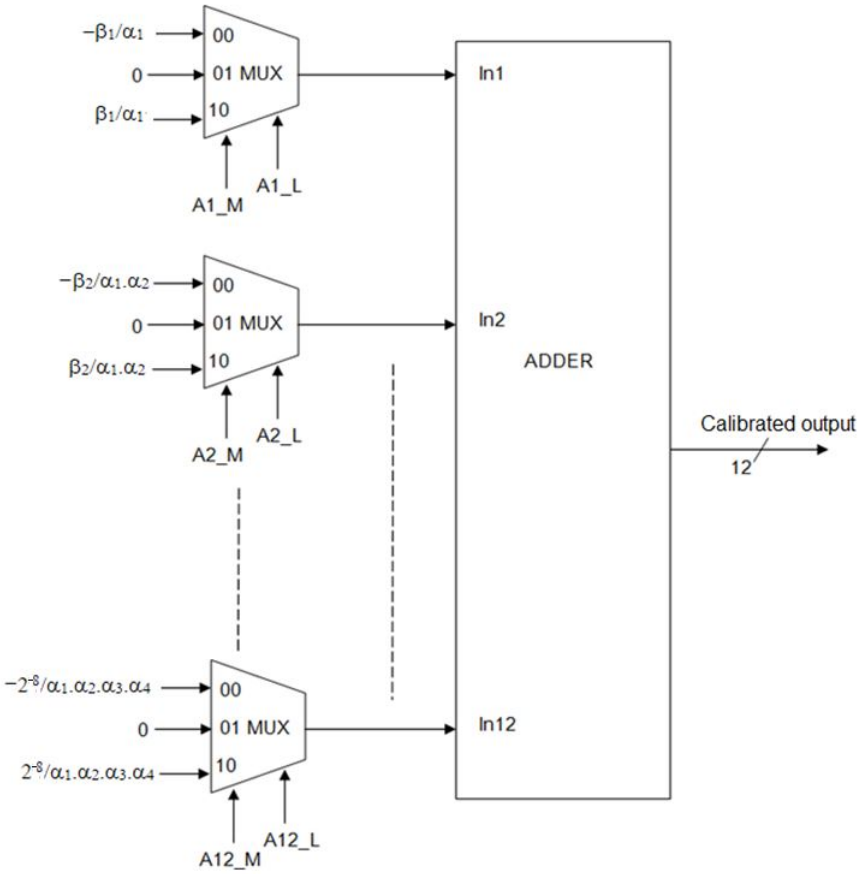

Figure FC7.4: 12 bit calibrated digital output of pipelined ADC

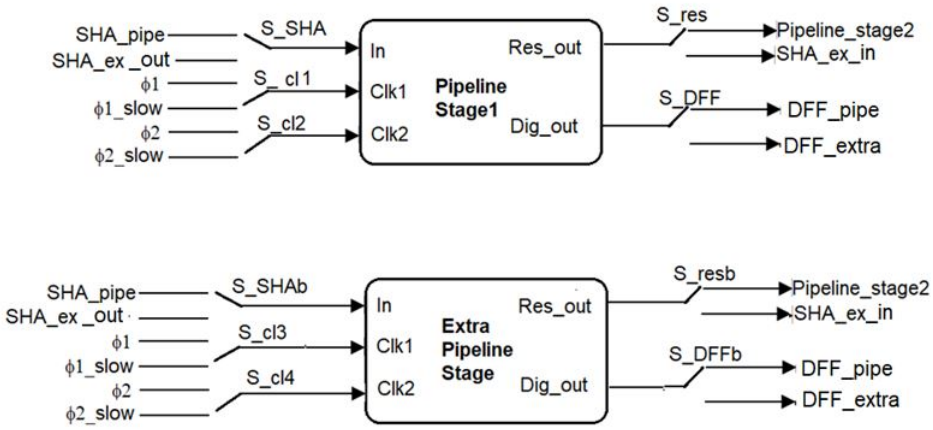

Figure FC7.5: Block diagram representing the calibration signals for calibrating the MSB stage

(*stage_3*) and pipeline stage4 (*stage_4*). Signals are assigned for calibration of each of these stages of the pipeline ADC. Let the signals to initiate calibration of *stage_ex*, *stage_1*, *stage_2*, *stage_3* and *stage_4* be $calex, cal1, cal2, cal3$ and $cal4$ respectively. The early calibration signals $cal1\_e, cal2\_e, cal3\_e$ and $cal4\_e$ are used to connect the high speed clock signals to the extra stage before initiation of calibration. The early calibration signals are raised one cycle before the calibration is initiated for the reasons explained in section 2. The extra pipeline stage is calibrated initially by raising the signal *calex* high. Once *stage_ex* is calibrated, the signal *calex* goes low. During calibration of *stage_1* of the pipelined ADC, it is required to change the clock signal of *stage_ex* from slow clock to high speed clock one cycle before it replaces *stage_1*. Also, it is required to swap the input output signals of *stage_1* with *stage_ex* at the time of replacement. The representative block diagram of this is as shown in Figure FC7.5. In Figure FC7.5, the signal *SHA_pipe* stands for the output of the SHA of the pipeline ADC, *SHA_ex_out* and *SHA_ex_in* denotes the output and input of the extra SHA required for calibration. $\phi 1$ and $\phi 2$ are high speed, non-overlapping clock signals required for the operation of the ADC. $\phi 1\_slow$ and $\phi 2\_slow$ are low speed, non-overlapping clock signals required during the calibration. $Pipeline\_stage2$ represents the input of the 2^{nd} stage of the pipeline ADC. DFF_pipe and DFF_extra represents the input to the D flip flop array which are connected at the output of $pipeline\_stage1$ and extra pipeline stage respectively. The connections to the switches *S_SHA*, *S_SHAb*, S_cl1, S_cl2, *S_res*, *S_resb*, S_DFF, S_DFFb, S_cl3 and S_cl4 during the normal operation of the ADC and during calibration are as shown in Table TC7.2 and Table TC7.3 respectively.

Table TC7.2: Switch connections during the normal operation of the ADC, when no pipe line stages are undergoing calibration

Switches	**Signals**
S_SHA	*SHA_PIPE*
S_cl1	$\phi 1$
S_cl2	$\phi 2$
S_res	$Pipeline\_stage2$
S_DFF	DFF_pipe2

During the normal operation of the ADC, when none of the stages are undergoing calibration, the extra pipeline stage is not functioning and hence the inputs and outputs of this stage are grounded.

Table TC7.3: Switch connections during calibration of the ADC, when pipeline stage1 undergoing calibration

Switches	**Signals**
S_SHA	SHA_ex_out
S_cl1	$\phi 1\_slow$
S_cl2	$\phi 2\_slow$
S_res	SHA_ex_in
S_DFF	DFF_extra
S_SHAb	SHA
S_cl3	$\phi 1$
S_cl4	$\phi 2$
S_resb	$Pipeline\_stage2$
S_DFFb	DFF_pipe

During the calibration of $stage\_1$, initially, $cal1\_e$ signal goes high. This indicates that the switches S_cl3 and S_cl4 are connected to $\phi 1$ and $\phi 2$ respectively. This switching activity is done before switching the inputs and outputs since $stage\_ex$ is switching to a high speed clock. In the subsequent clock cycle, $cal1$ signal goes high and the signals are connected to the switches as shown in Table TC7.3. This arrangement takes care of the normal working of the pipeline ADC. Now, the digital calibration coefficients α_1 and β_1 are calculated as discussed in section 2. Subsequent to calibration of $stage\_1$, $Cal1\_e$ signal goes low indicating that the fast clock signals are connected to the MSB pipeline stage. Then, $Cal1$ signal goes low indicating that MSB pipeline stage replaces the extra pipeline stage in the pipeline ADC with connections as shown in Table TC7.2. α_1 and β_1 of the MSB stage calculated during calibration are stored in the memory to be used by the pipeline ADC. During the calibration of $stage\_2$, $cal2\_e$ signal goes high indicating that the switches S_cl3 and S_cl4 are connected to $\phi 2$ and $\phi 1$ respectively, during replacement. This is due to the fact that the pipeline ADC works in a ping pong fashion. Note that two calibration signals, (eg. $Cal1$ and $Cal2$) cannot

be high at the same time as two stages cannot be calibrated at the same time using this algorithm. When *stage_1* is replaced by *stage_ex* for calibration, *stage_ex* now forms the MSB pipeline stage. During the period, α and β of the *stage_ex* are used during data conversion by the pipelined ADC in the place of α_1 and β_1 of *stage_1*. This holds well during calibration of any stage.

7.2.3 Calibrating two stages at a time

The advantage of calibrating 2 stages at a time is that it neither requires an ESHA nor swapping clock signals. Therefore, this technique does not require early calibration signals. Let the calibration signals for calibrating *stage_ex*1 and *stage_ex*2 be *calex*, *stage_1* and *stage_2* be *cal*12, *stage_3* and *stage_4* be *cal*34 respectively. Signals *swap*1 and *swap*2 are used to swap *stage_1* and *stage_2* with *stage_ex*1 and *stage_ex*2 during calibration. Similarly, signals *swap*3 and *swap*4 are used to swap *stage_3* and *stage_4* with *stage_ex*1 and *stage_ex*2 during calibration. Figure FC7.6 shows a representation of the signals required for calculation of calibration coefficients of the 2 MSB pipeline stages. The signal *SHA_pipe* represents the output of the SHA of the pipeline ADC. *Calib_signal* is the $\frac{V_{ref}}{4}$ voltage required for calibration. *Pipe*1*_res* and *Pipe*2*_res* correspond to the residue voltage of the 1^{st} and 2^{nd} MSB pipeline stage of the pipeline ADC. *Pipe*1*_res_ex* and *Pipe*2*_res_ex* signify the residue voltage of the 1^{st} and 2^{nd} extra pipeline stages required during calibration. *Pipeline_stage*3 represent the input of the 3^{rd} MSB stage of the pipeline ADC. *DFF_pipe*1 and *DFF_pipe*2 denotes the input to the D flip flop array connected from the 1^{st} and 2^{nd} MSB pipeline stages respectively. Similarly *DFF_pipe_ex*1 and *DFF_pipe_ex*2 denotes the input to the D flip flop array connected from the 1^{st} and 2^{nd} extra pipeline stages respectively. Initially, the extra stages are calibrated by raising *calex* signal high. The connections to the switches *S_in*1, *S_in*2, *S_in*1*b*, *S_in*2*b*, *S_res*, *S_resb*, *S_DFF*1, *S_DFF*2, *S_DFF*3 and *S_DFF*4 during the normal operation of the ADC and during calibration are as shown in Table

TC7.4 to Table TC7.7 respectively.

Table TC7.4: Switch connections during the normal operation of the ADC, when none of the calibration signals are high

Switches	**Signals**
S_in1	SHA_pipe
S_in2	$Pipe1\_res$
S_res	$Pipeline\_stage3$
S_DFF2	DFF_pipe2
S_DFF1	DFF_pipe1
S_in1b	-
S_in2b	-
S_resb	-
S_DFF3	-
S_DFF4	-

Table TC7.5: Switch connections when the signal $cal1$ is high and $cal2$ is low

Switches	**Signals**
S_in1	$Calib\_signal/Pipe2\_res$
S_in2	$Pipe1\_res\_ex$
S_res	$Pipeline\_stage3$
S_DFF2	DFF_pipe2
S_DFF1	DFF_pipe_ex1
S_in1b	SHA_pipe
S_in2b	-
S_resb	-
S_DFF3	DFF_pipe1
S_DFF4	-

Table TC7.6: Switch connections when both signals cal1 and cal2 are high

Switches	**Signals**
S_in1	$Calib\_signal/Pipe2\_res$
S_in2	$Calib\_signal/Pipe1\_res$
S_res	$Pipe2\_res$
S_DFF2	DFF_pipe_ex2
S_DFF1	DFF_pipe_ex1
S_in1b	SHA_pipe
S_in2b	$Pipe1\_res\_ex$
S_resb	$Pipeline\_stage3$
S_DFF3	DFF_pipe1
S_DFF4	DFF_pipe2

Raising signals $swap1$ and $swap2$ high, indicates that the input output signals of

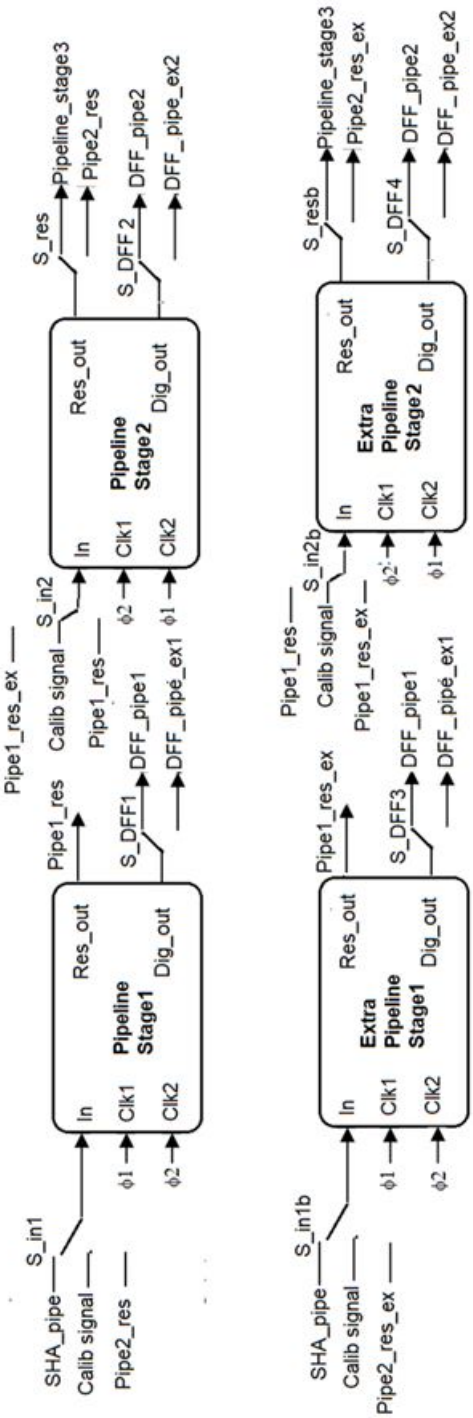

Figure FC7.6: Block diagram representing the calibration signals for calibrating two MSB stages

Table TC7.7: Switch connections when the signal $cal2$ is high and $cal1$ is low

Switches	**Signals**
S_in1	SHA_pipe
S_in2	-
S_res	-
S_DFF2	-
S_DFF1	DFF_pipe_1
S_in1b	-
S_in2b	$Pipe1\_res$
S_resb	$Pipeline\_stage3$
S_DFF3	-
S_DFF4	DFF_pipe2

$pipeline\_stage1$ and $pipeline\_stage2$ are swapped with $pipeline\_stage\_ex1$ and $pipeline\_stage\_ex2$ stages respectively. As explained in section 2, $swap1$ is raised just before the raising edge of $\phi 1$. Similarly, $swap2$ is raised just before the raising edge of $\phi 2$. With the arrangement as shown in Table TC7.6, the 2 most significant pipeline stages are ready to undergo calibration. Since, the extra pipeline stages that are already calibrated replace the pipeline stages at appropriate time; the normal working of the ADC is not affected. Subsequent to this, $cal12$ signal is raised to indicate the initiation of calibration of the 2 stages. The switch S_in1 is connected to $Calib$ signal while sampling the signal $\frac{V_{ref}}{4}$ and to $Pipe2\_res$ for the rest of the cycles. Similarly, the switch S_in2 is connected to $Calib$ signal while sampling the signal $\frac{V_{ref}}{4}$ and to $Pipe1\_res$ for the rest of the cycles. Calibration of 2 stages takes place as explained in section 2. Post calibration, $cal12$ signal goes low indicating the end of calibration of the 2 stages. $Swap1$ and $Swap2$ goes low just before the raising edge of $\phi 1$ and $\phi 2$ respectively indicating that pipeline stage1 and pipeline stage2 takes its place in the pipeline ADC. This is as shown in Figure FC7.6. Similar to calibrating one stage at a time, it is to be noted that during the calibration of pipeline stage1 and pipeline stage2, the extra pipeline stages replace these stages. Hence, during this time, α and β of the extra pipeline stages are used to obtain the calibrated digital code of the pipelined ADC.

7.3 Simulation Results

The 12 bits pipeline ADC which consists of 11 pipeline stages and one 2 bits flash ADC (Appendix A.6) as well as the extra pipeline stages are designed and implemented in a 180-nm CMOS technology of Semiconductor Complex Limited, India (SCL), and were simulated in Cadence-AMS environment.

The 12 bits pipelined ADC was designed to handle inputs ranging from -1.2V to 1.2V. The proposed calibration algorithms were tested using a ramp and sine wave input. Figure FC7.7 shows a graph of output characteristics of the ADC for a ramp input varying between -400mV to 500mV. Up to 6.25% capacitance mismatches were introduced in the 2 MSB stages to validate the proposed algorithm calibrating one stage at a time. This holds good for calibrating the 3^{rd} and 4^{th} pipeline stages as well. The output digital bits of the pipeline ADC is converted into its equivalent analog value for ease of visualization. As discussed in section 2, it can be seen that the extra pipeline stage is calibrated in the beginning followed by first stage and then the second stage. Jumps due to capacitor mismatch in the first stage and second stage can be seen at input voltages close to -300mV and -150mV. These jumps in the output characteristic rests in the fact that the first pipeline stage and second pipeline stage are not calibrated yet. These are marked in red circles in Figure FC7.7. The green, orange and blue rectangles depict the period during which the extra pipeline stage, first pipeline stage and second pipeline stage undergo calibration respectively. Jump at +150mV is not seen in the output characteristic due to the fact that the 2^{nd} pipeline stage is replaced by the already calibrated extra stage. Subsequent to calibration, ADC output characteristics vary linearly with respect to the inputs without resulting in any jumps. The effective functioning of the pipeline ADC is not hampered during calibration of the pipeline stages.

Similarly, FC7.8 shows the output characteristics of the ADC during calibration of 2

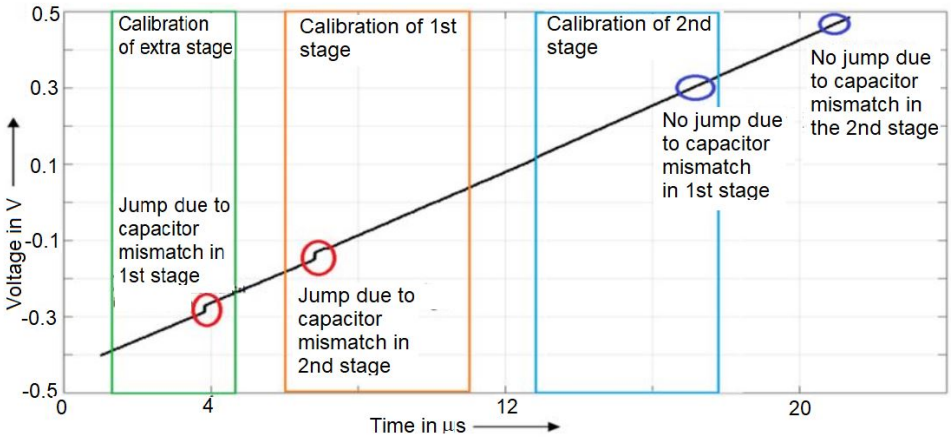

Figure FC7.7: Output characteristics of the pipeline ADC calibrating 2 MSB stages one stage at a time

stages at a time. Capacitance mismatch in the 2 MSB stages introduce jumps at -300mV and -150mV as marked in red circles in Figure FC7.8. The green and grey rectangles depict the period during which the extra pipeline stages and 2MSB pipeline stages are calibrated respectively. Jump at +150mV is not seen in the output characteristic due to the fact that the 2^{nd} pipeline stage is replaced by the already calibrated extra stage. It can be seen that after the calibration of the 2 MSB stages, the jumps due to the capacitor mismatches are not seen in the output transfer characteristics of the ADC. These are marked in blue circles.

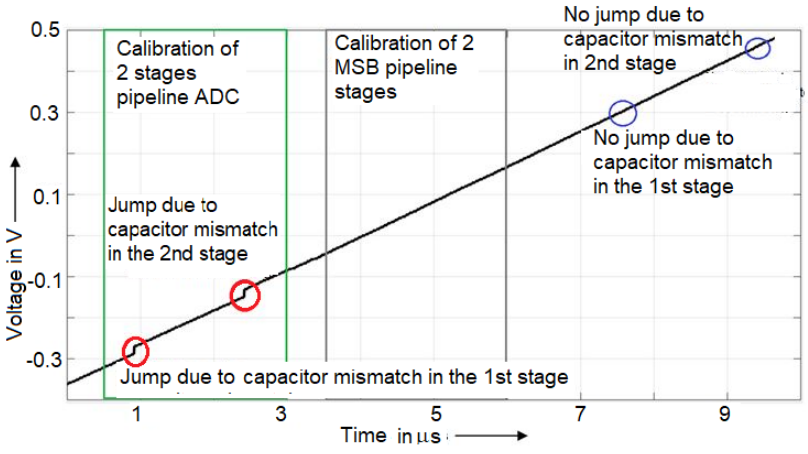

Figure FC7.8: Output characteristics of the pipeline ADC calibrating 2 MSB stages two stages at a time

Figure FC7.9 shows the INL plot of the pipeline ADC before and after calibration. Up to 5% capacitance mismatches were introduced to validate the proposed algorithm. A ramp input varying from -1.2V to 1.2V was input to the pipeline ADC. The output of the pipeline ADC was processed in Matlab to obtain the INL and DNL of the pipeline ADC. It is seen that INL and DNL of the ADC reduces to 0.71 LSB and 0.35 LSB respectively after the calibration of the ADC one stage at a time and to 0.69 LSB and 0.34 LSB respectively after the calibration of the ADC two stages at a time. FC7.10

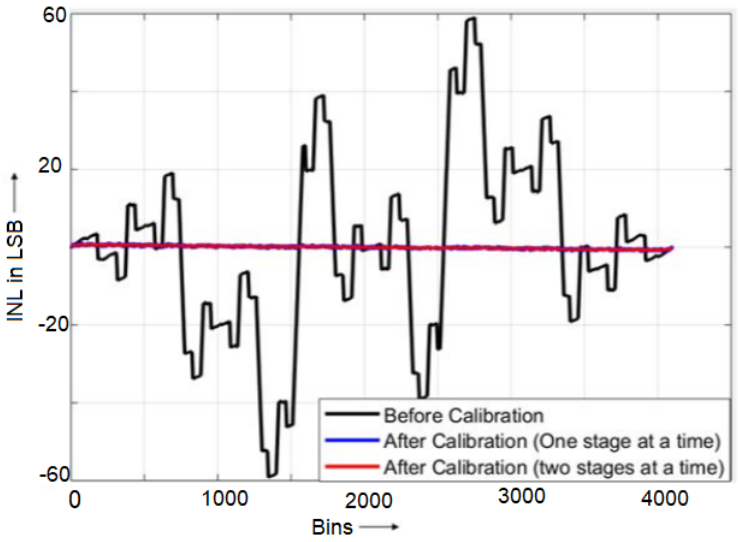

Figure FC7.9: Plot of INL of the ADC before and after calibration of ADC

shows the plot indicating the variations of SNR and SFDR of the ADC before and after calibration for up to 6.25% capacitance mismatch. SNR-BC and SFDR-BC represent the SNR and SFDR of the ADC before calibration. SNR-AC1, SNR-AC2, SFDR-AC1 and SFDR-AC2 represent the SNR and SFDR of the ADC after calibration using Technique 1 and Technique 2 respectively. Significant improvement in SNR and SFDR is observed after the calibration of the ADC using the proposed techniques. FC7.11 shows the output spectrum of a 14MHz full scale analog sine input which is sampled at 40 MSPS by the pipelined ADC, before and after calibration. It is seen that SFDR improves from 36 dB before calibration to 69 dB after calibration while calibrating one stage at a time and to 70 dB after calibration while calibrating two stages at a time.

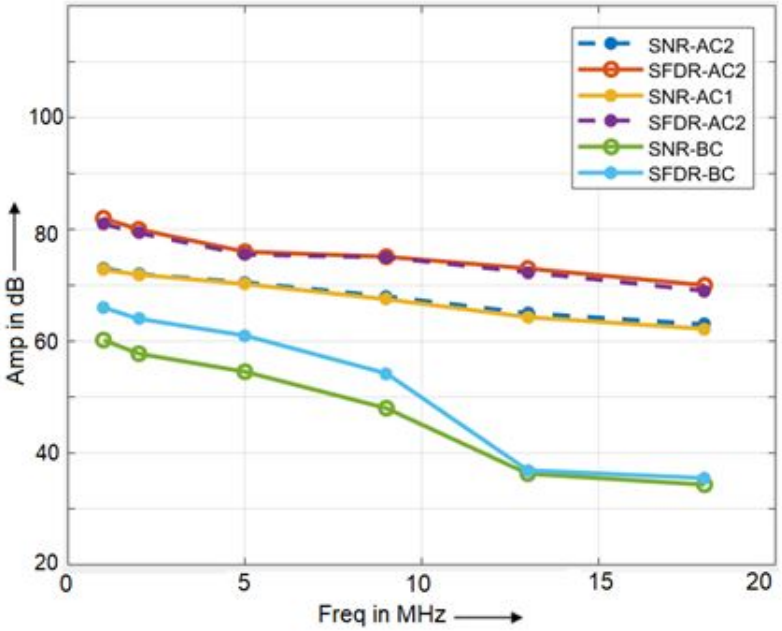

Figure FC7.10: Plot of SNR and SFDR before and after calibration of ADC

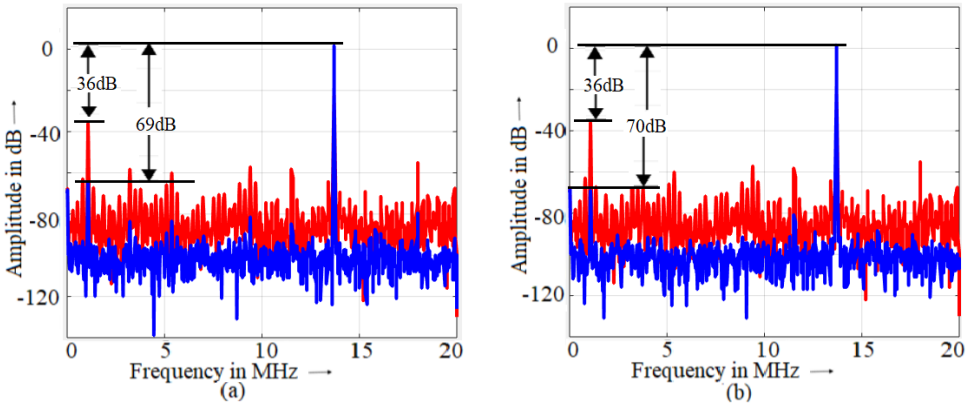

Figure FC7.11: FFT plot of the ADC Before calibration and After calibration (a) using technique 1 (b) using technique 2

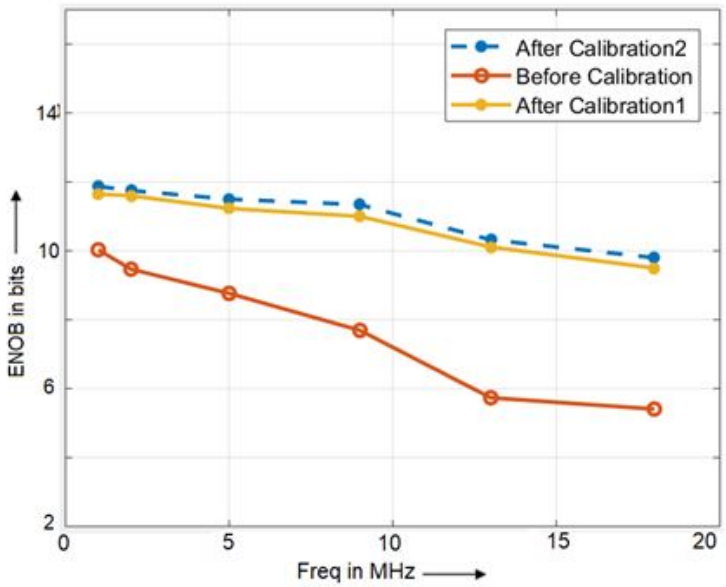

Figure FC7.12: ENOB plot of the ADC before and after calibration

Figure FC7.12 represents the plot of Effective Number Of Bits (ENOB) before calibration and after calibration of the ADC. Table TC7.8 summarizes the performance of the ADC after implementing the calibration algorithms.

7.4 Conclusion

A novel background deterministic calibration technique to calibrate a pipelined ADC is presented in this chapter. The calibration algorithm has been tested on 12 bits, 1.5 bits/stage pipelined ADC with finite op-amp gain and capacitor mismatch of up to 6.25%. The advantage of the proposed technique is that the calibration algorithm is deterministic and hence takes very few clock cycles for calibration. It is seen that the INL of the calibrated pipeline ADC was reduced to 0.71 LSB and 0.69 LSB when calibrated using technique 1 and technique 2 respectively. A significant improvement in the SNR, ENOB and SFDR of the ADC was also observed.

Table TC7.8: Performance table of the pipelined ADC after calibration

	Calibrating 1 stage at a time	Calibrating 2 stages at a time
Resolution	12-bits	12-bits
Sampling rate	40MHz	40MHz
SNR	64.1dB@14MHz, 73 dB @ 1MHz	63.32 dB@14MHz, 72.67 dB @ 1MHz
SFDR	70 dB@14MHz, 82 dB@ 1MHz	69 dB@14MHz, 81 dB@ 1MHz
INL	0.69LSB	0.71LSB
DNL	0.34LSB	0.35LSB
Input voltage swing	1.2Vp-p	1.2Vp-p
Power dissipation (Without scaling currents in stages)	60mW	58mW
Power dissipated by the digital calibration hardware	4.18mW	3.34mW
Process	1.8V, 180nm CMOS	1.8V, 180nm CMOS
Power supply	1.8V	1.8V

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

This chapter summarizes the conclusions of this thesis and discusses the areas for future work.

8.1 Conclusions

This thesis presents 6 novel calibration techniques to calibrate the pipelined ADC out of which 4 are foreground and 2 background calibration techniques. The goal of the thesis is to calibrate the pipelined ADCs which suffer from non linearity due to mismatches in the analog components like capacitance mismatch, finite op-amp gain error, parasitic capacitance and comparator offsets. All the algorithms were initially implemented in MATLAB to verify the functionality of the calibration algorithm. The pipelined ADC and algorithmic ADC were implemented in a 180-nm CMOS technology of Semiconductor Complex Limited, India (SCL), and were simulated in Cadence-AMS environment with the digital portions coded using Verilog HDL.

The first calibration technique discussed in chapter 3 accounts for the errors introduced due to small capacitance mismatches in the MX2 stage. The technique is characterized by its simplicity, minimal hardware and low power dissipation. The second calibration technique discussed in chapter 4 calibrates the errors due to capacitance

mismatches, finite op-amp gain and parasitic capacitances. The above techniques can be applied to any 1.5 bits/stage pipeline ADC if there is an access to intermediate digital codes. The pipeline hardware is neither interrupted nor externally controlled during calibration. The third calibration technique discussed in chapter 5 calibrates the errors due to capacitance mismatches, finite op-amp gain and parasitic capacitors. The calibrations coefficients of all the stages are accurate to 12 bits. The technique requires another slow and accurate reference ADC for calibration. The forth calibration technique discussed in chapter 6 calibrates the errors due to capacitance mismatches, finite op-amp gain, offset errors and parasitic capacitors. The calibration coefficients of all the stages are calculated from 11 backend stages and are hence accurate unlike the 2^{nd} calibration technique. The accuracy of the calibration coefficients are maintained without using a reference ADC. Two background calibration techniques that accounts for the errors due to capacitance mismatches, finite op-amp gain and parasitic capacitors are discussed in chapter 7. The calibration coefficients are calculated to 12 bits accuracy in background and are utilized to obtain calibrated digital output during the normal operation of the ADC.

Table TC8.1 gives a comparison all the calibration techniques that are discussed in the thesis.

Table TC8.1: Comparison of the proposed calibration techniques

Technique	1	2	3	4	5	6
Calibration	Foreground	Foreground	Foreground	Foreground	Background	Background
Accounts for	Small capacitance mismatches in MX2 stage	Capacitance mismatch, finite op-amp gain and parasitic capacitors	Capacitance mismatch, finite op-amp gain and parasitic capacitors	Capacitance mismatch, finite op-amp gain, offset in the MX2 amplifier and parasitic capacitors	Capacitance mismatch, finite op-amp gain and parasitic capacitors	Capacitance mismatch, finite op-amp gain and parasitic capacitors
Extra analog hardware	Modify SHA to work as ramp generator	Modify SHA to work as ramp generator	Modify SHA to work as ramp generator, Requires a slow and accurate reference ADC	$\frac{V_{ref}}{4}$, -$\frac{V_{ref}}{4}$ reference generator	$\frac{V_{ref}}{4}$, -$\frac{V_{ref}}{4}$ reference generator and an extra SHA and MX2 stage	$\frac{V_{ref}}{4}$, -$\frac{V_{ref}}{4}$ reference generator and 2 extra MX2 stages
Speed of calibration	slow	moderate	Very slow	Moderate	fast	fast
Resolution in bits	9	9	12	12	12	12
SNDR in dB	50	51	62	62	64	63
INL in LSB	0.72	0.66	0.65	0.71	0.67	0.65
DNL in LSB	0.56	0.4	0.3	0.29	0.34	0.34
Power dissipated by the digital hardware in Watts	76u	460u	220u	3.56m	4.18m	3.3m

8.2 Future scope

All the calibration techniques have been simulated at the schematic level in cadence. The next step would be to perform RC extraction by laying out the transistors and validate the results.

The calibration techniques mentioned above does not take care of the 3^{rd} order non-linearity of the op-amp. The calibration techniques can be extended to account for 3^{rd} order non linearity as well.

Perform Monte-Carlo simulations to study the impact of process variations on the proposed calibration techniques.

Bibliography

[1] R. Jacob Baker. *CMOS Circuit Design Layout and Simulation*. WILEY, third edition, 2010. doi: 10.1002/9780470891179.

[2] Chithira Ravi. *Improvised Calibration Techniques for High Speed Pipelined ADC*. PhD thesis, Amrita Vishwa Vidyapeetam, Amritpuri Campus (INDIA), 2018.

[3] A.N. Karanicolas, H.S. Lee, and K.L. Bacrania. A 15-b 1-Msample/s Digitally Self-calibrated Pipelined ADC. *IEEE J. Solid-State Circuits*, 28(12):1207–1215, 1993. doi: 10.1109/4.261994.

[4] Bibhu Datta Sahoo and Behzad Razavi. A 10-b 1-GHz 33-mW CMOS ADC. *IEEE J. Solid-State Circuits*, 48(6):1442–1452, 2013. doi: 10.1109/JSSC.2013.2252518.

[5] Shang-Yuan (Sean) Chuang and Terry L. Sculley. A Digitally Self-Calibrating 14-bit 10-MHz CMOS Pipelined A/D Converter. *IEEE J. Solid-State Circuits*, 37(6):674–683, 2002. doi: 10.1109/JSSC.2002.1004571.

[6] Bibhu Datta Sahoo and Behzad Razavi. A 12-Bit 200-MHz CMOS ADC. *IEEE J. Solid-State Circuits*, 44(9):2366–2380, 2009. doi: 10.1109/JSSC.2009.2024809.

[7] Chithira Ravi, Vineeth Sarma, and Bibhu Datta Sahoo. At speed digital gain error calibration of pipelined adcs. In *2015 IEEE 13th International New Circuits and Systems Conference (NEWCAS)*, pages 100–110, Apr. 1996. doi: 10.1109/NEWCAS.2015.7182041.

[8] Vineeth Sarma, Chithira Ravi, and Bibhu Datta Sahoo. Achieving Theoretical Limit of SFDR in Pipelined ADCs. *IEEE J. Solid-State Circuits*, 25(11):3175–3185, 2017. doi: 10.1109/TVLSI.2017.2731262.

[9] Yun-Shiang Shu and Bang-Sup Song. A 15-bit Linear 20-MS/s Pipelined ADC Digitally Calibrated With Signal-Dependent Dithering. *IEEE J. Solid-State Circuits*, 43(2):342–350, 2008. doi: 10.1109/JSSC.2007.914260.

[10] C.R. Grace, P.J. Hurst, and S.H. Lewis. A 12-bit 80-MSample/s pipelined ADC with bootstrapped digital calibration. *IEEE J. Solid-State Circuits*, 40(5):1038–1046, 2005. doi: 10.1109/JSSC.2005.845972.

[11] Xiaoyue Wang, P.J. Hurst, and S.H. Lewis. A 12-bit 20-Msample/s pipelined analog-to-digital converter with nested digital background calibration. *IEEE J. Solid-State Circuits*, 39(11):1799–1808, 2004. doi: 10.1109/JSSC.2004.835826.

[12] Y. Chiu, C. W. Tsang, B. Nikolic, and P. R. Gray. Least mean square adaptive digital background calibration of pipelined analog-to-digital converters. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 51(1):38–46, 2004. doi: 10.1109/TCSI.2003.821306.

[13] Nan Sun, Hae-Seung Lee, and Donhee Ham. Digital Background Calibration in Pipelined ADCs using Commutated Feedback Capacitor Switching. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 55(9):877–881, 2008. doi: 10.1109/TCSII.2008.924369.

[14] O.E. Erdogan, P.J. Hurst, and S.H. Lewis. A 12 b digital-background-calibrated algorithmic ADC with -90 dB THD. *IEEE J. Solid-State Circuits*, 34(12):1812 – 1820, 1999. doi: 10.1109/4.808906.

[15] Jipeng Li and Un-Ku Moon. Background Calibration Techniques for Multistage Pipelined ADCs with Digital Redundancy. *IEEE Transactions on Circuits and*

Systems II: Analog and Digital Signal Processing, 50(9):531 – 538, 2003. doi: 10.1109/TCSII.2003.816921.

[16] H. Adel, M.-M. Louerat, and M. Sabut. Fast Split Background Calibration for Pipelined ADCs enabled by Slope Mismatch Averaging Technique. *Electronics Letters*, 48(6):318 – 320, 2012. doi: 10.1049/el.2012.0357.

[17] Omran. H., Alahmadi. H, and Salama. K. N. Matching properties of femtofarad and sub-femtofarad mom capacitors. *IEEE Transactions on Circuits and Systems-I*, 63(6):763–772, 2016. doi: 10.1109/TCSI.2016.2537824.

[18] Chinmaye. Ramamurthy, Surya. Padma, Subhajit. S, and Chetan Parikh. A Deterministic Digital Calibration Technique for Pipelined ADCs using a Non-nested Algorithm. *Analog IC design and Signal processing*, 110:557–568, 2022. doi:10.1007/s10470-021-01961-5.

[19] B. Razavi. The StrongARM Latch [A Circuit for All Seasons]. *IEEE Solid-State Circuits Magazine*, 7:12 – 17, 2015. doi:10.1109/MSSC.2015.2418155.

[20] Andrew Masami Abo. *Design for reliability of low-voltage, switched-capacitor circuits*. PhD thesis, University of California, Berkley, 1999.

[21] B Razavi. *Design of Analog CMOS Integrated Circuits*. Tata McGraw-Hill Publishing Company Limited, third edition, 2001.

[22] P. Surya, S. Das, S. Sen, and C. Parikh. High Performance Operational Amplifier with 90dB Gain in SCL 180nm Technology. In *24th International Symposium on VLSI Design and Test (VDAT)*, July. 2020. doi: 10.1109/VDAT50263.2020.9190288.

[23] Chinmaye. Ramamurthy, Subhajit. S, and Chetan Parikh. Deterministic Digital Calibration Technique for 1.5 bits/stage Pipelined and Algorithmic ADCs with

Finite Op-Amp gain and Large Capacitance Mismatches. *Circuits, Systems and signal processing*, 40:3684–3702, 2021. doi: 10.1007/s00034-021-01652-6.

[24] Chinmaye. Ramamurthy, Subhajit. S, and Chetan Parikh. Digital Calibration of 1.5 bits/stage Algorithmic ADC. In *18th International SoC Design Conference*, Oct 2021. doi: 10.1109/ISOCC53507.2021.9614017.

[25] W. Li. P, Chin. M. J, and Castello. R. A ratio-independent algorithmic analog-to-digital conversion technique. *IEEE Journal of Solid-State Circuits*, 19(6):828–836, 1984. doi: 10.1109/JSSC.1984.1052233.

[26] Rebentrost. P, Schuld. M, Wossnig. L., F. Petruccione, and Lloyd. S. Quantum gradient descent and newton's method for constrained polynomial optimization. *New Journal of Physics*, 21, 2019. doi: 10.1088/1367-2630/ab2a9e.

[27] Chinmaye. Ramamurthy, Subhajit. S, and Chetan Parikh. Deterministic Digital Calibration of 1.5 bits/stage Pipelined ADCs by Direct Extraction of Calibration Coefficients. In *34th International Conference on VLSI Design*, Feb 2021. doi: 10.1109/VLSID51830.2021.00010.

[28] J. D. Faires Richard.L.Burden. *Numerical Analysis, Theory and Applications*. CENGAGE Learning, ninth edition, 2010.

APPENDIX A

CIRCUITS USED IN PIPELINE ADC

A.1 High gain Op-amp

The op-amp works rail-to-rail, and has been designed with a telescopic cascode stage followed by a push-pull common-source stage [22]. The gain achieved by the op-amp is 87 dB with a unity gain frequency of 500 MHz and a phase margin greater than 56^o over all process corners. FA1.1 presents the complete circuit diagram of the high gain opamp, which includes a telescopic cascode differential amplifier in the first stage, its CMFB circuit, stage-2 with its common-mode feedback (CMFB), and two separate dc bias circuits. RC-Miller compensation is used for the main opamp. Two separate dc biasing networks are designed: the one labelled "Bias Network" in FA1.1 supplies biases to all sub-networks except the first-stage telescopic cascode differential amplifier. The latter has its own bias network, which labelled "Cascode Bias" in FA1.1. The detailed descriptions of the individual blocks are discussed in the following sub-sections.

A.1.1 Bias Circuit for Input Stage

The bias circuit implemented generates only two bias voltages (V_up and V_down). The voltage V_up is used to bias M3-M4 as well as M7-M8. This is possible because

the swing required at the output of the first stage is very small. This is compensated by the gain of the second stage (around 30dB). Therefore, the same DC voltages can be used at M3-M4 and M7-M8, to keep all transistors in saturation.

A.1.2 Second Stage

FA1.2 shows source degenerated class AB stage. The stage of FA1.2(b) provides a push-pull operation at the output node, which increases the gain as well as the slew rate, compared to the traditional CS stage of FA1.2(a). The source degenerated configuration is chosen because it provides an additional degree of freedom to control the dc output voltages ($StII\_OutP$, $StII\_OutN$) by controlling the gate voltage of M11 via a CMFB loop. This is necessary for setting the output voltages to a dc value of zero. The push-pull circuit creates an additional pole at the drain of $M9_i$ to optimize the location of this pole, the left half of FA1.2(c)) is designed to carry $\frac{1}{8}^{th}$ the magnitude of the current in the right half. This also makes the first stage experience a comparatively smaller load. By using the class AB stage, the overall gain of the opamp increased from 81dB to 89dB.

A.1.3 Common-Mode Feedback (CMFB)

Changes in the common-mode input voltage can cause the outputs of the second stage to saturate, even for a small first stage common-mode gain, due to the gain of the second stage. Thus, CMFB is necessary for the first stage to significantly reduce its common-made gain. CMFB is needed for the second stage to ensure a well-defined dc output voltage. A continuous time CMFB configuration is used in the design. Discrete time CMFB circuits involve additional capacitors and switches thereby increasing the load capacitance and reducing the overall speed.

The circuit was implemented in a 180-nm CMOS technology of Semiconductor

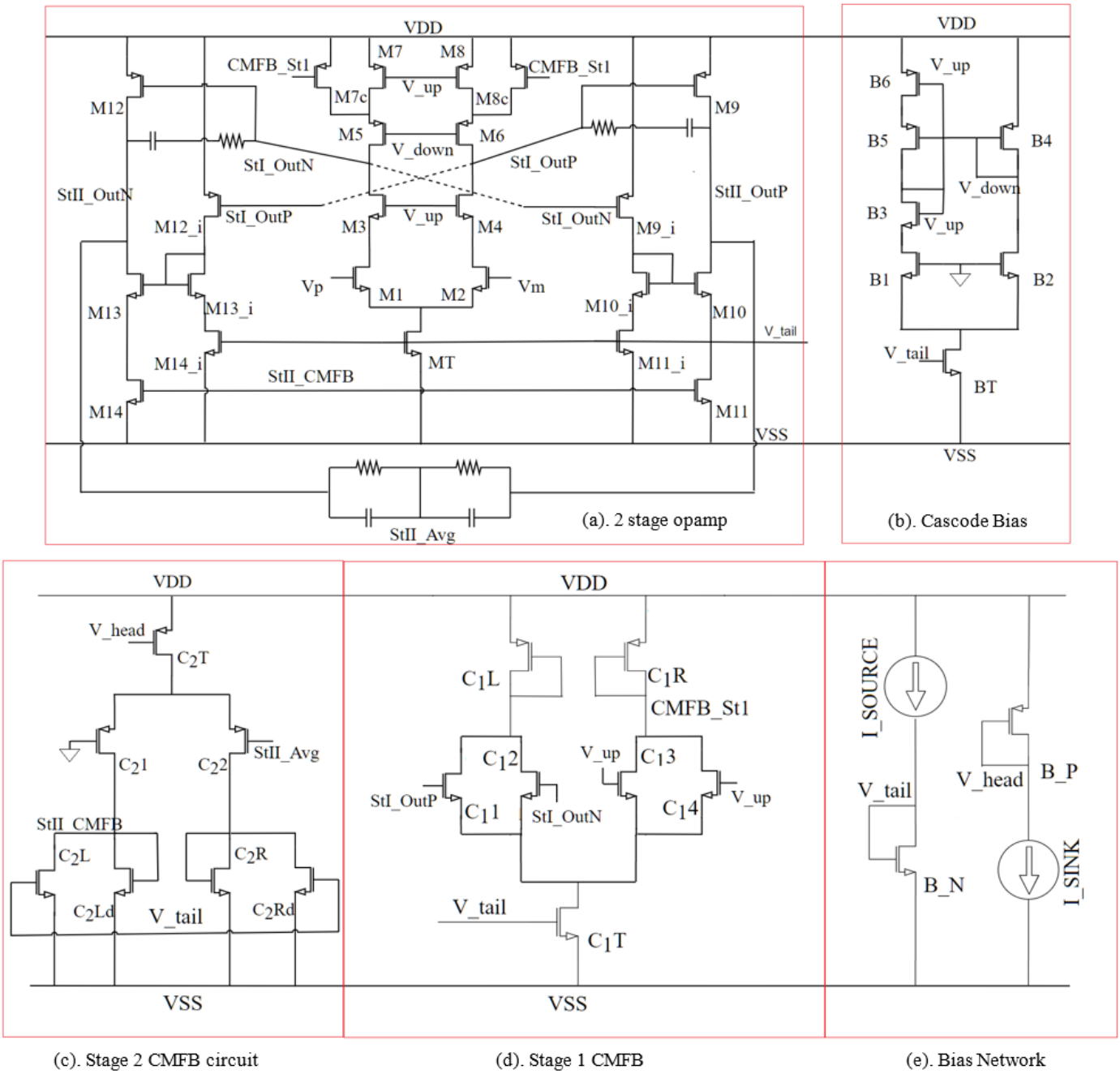

Figure FA1.1: High gain Op-amp

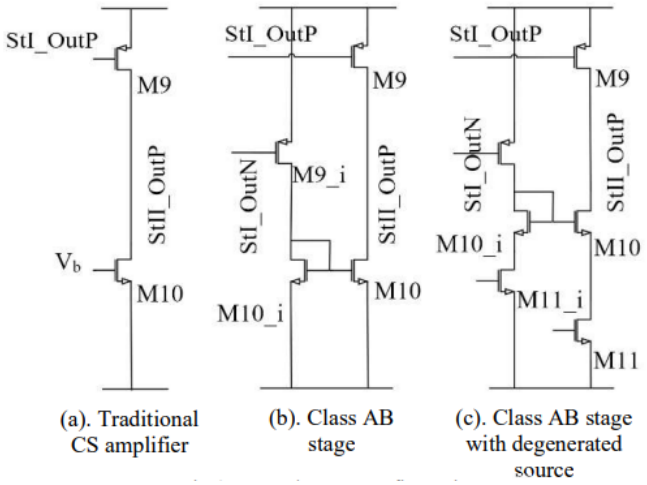

Figure FA1.2: Stage 2 configurations

Complex Limited (SCL), India. Simulations were done in Cadence® environment. FA1.3 shows Bode plots of the overall opamp with a load capacitance of 1 pF. A dc gain of 89 dB, and a unity-gain bandwidth (UGB) of 584 MHz at a phase margin (PM) of 55° were obtained.

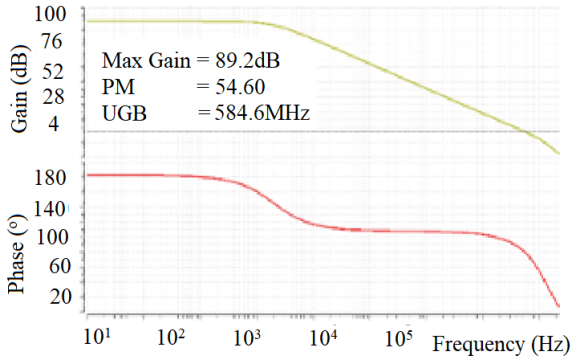

Figure FA1.3: High gain op-amp Bode plots (with a 1pF load)

FA1.4 shows the output spectrum (N=2048) of a sample and hold amplifier for 0.69MHz and 1.91MHz full scale analog sine input which is sampled at 4 MSPS by the op-amp.

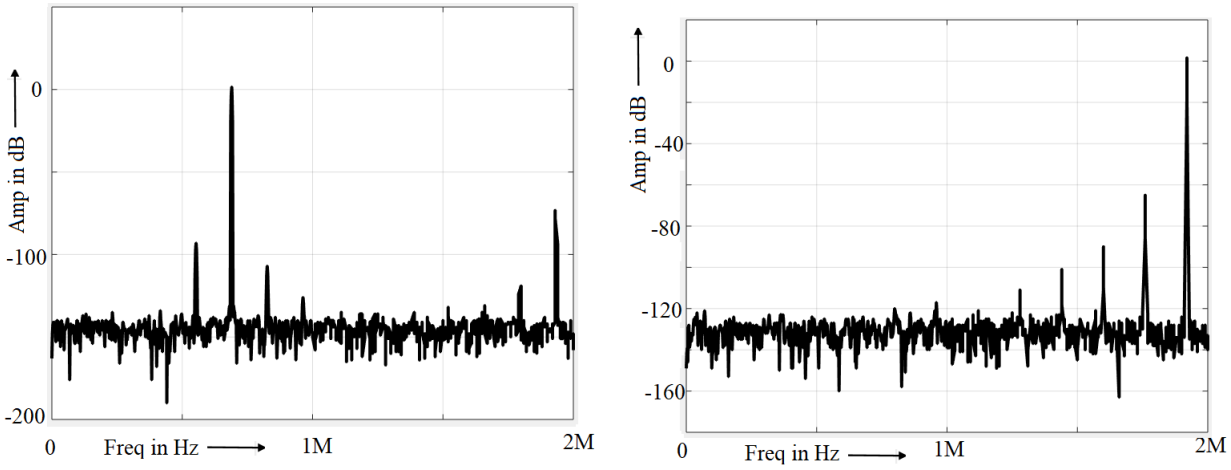

Figure FA1.4: Output spectrum of the op-amp (with a 1pF load)

A.2 Low gain two stage Op-amp

The op-amp has been designed with a 5T differential amplifier with NMOS input pair in the first stage followed by a common-source stage. This is as shown in FA1.5. The gain achieved by the op-amp is 52dB with a unity gain frequency of 700 MHz and a phase margin greater than 59^o. The op-amp works with the resistive CMFB circuit. The transistors M1, M2, M3, M4 and M9 form the first stage amplifier. M5, M6, M7 and M8 form the second stage amplifier. Resistors R1, R2, R5 and R6 form a CMFB circuit. Without transistor M10, the output common mode level at V_o is equal to the gate source voltage of M8. This voltage is slightly greater than the threshold voltage of M8 and far away from zero. In order to maintain the required common mode level at the output, a small transistor M10 is added so that it draws just enough current to maintain the output common mode level at the required value [21]. The op-amp works with supply voltage -0.9V and 0.9V and draws a total current of 1.6mA.

The circuit was implemented in a 180-nm CMOS technology of Semiconductor Complex Limited (SCL), India. Simulations were done in Cadence® environment. FA1.6 shows Bode plots of the overall opamp with a load capacitance of 1.2 pF. A dc

gain of 52 dB, and a unity-gain bandwidth (UGB) of 700 MHz at a phase margin (PM) of 59° were obtained.

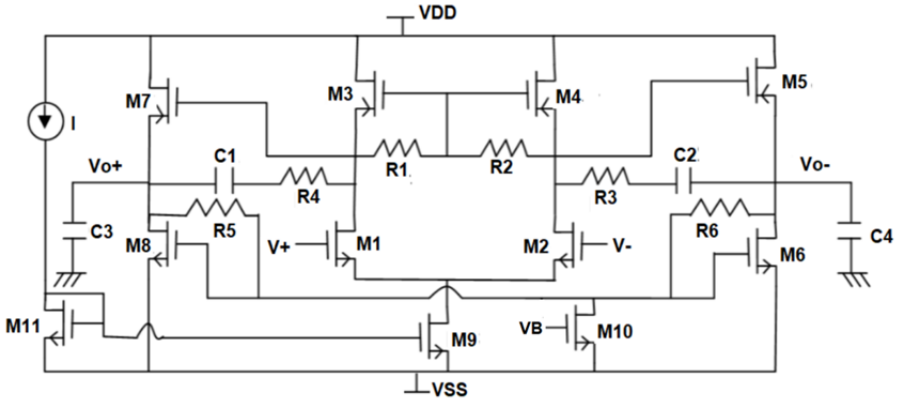

Figure FA1.5: Two stage low gain Op-amp

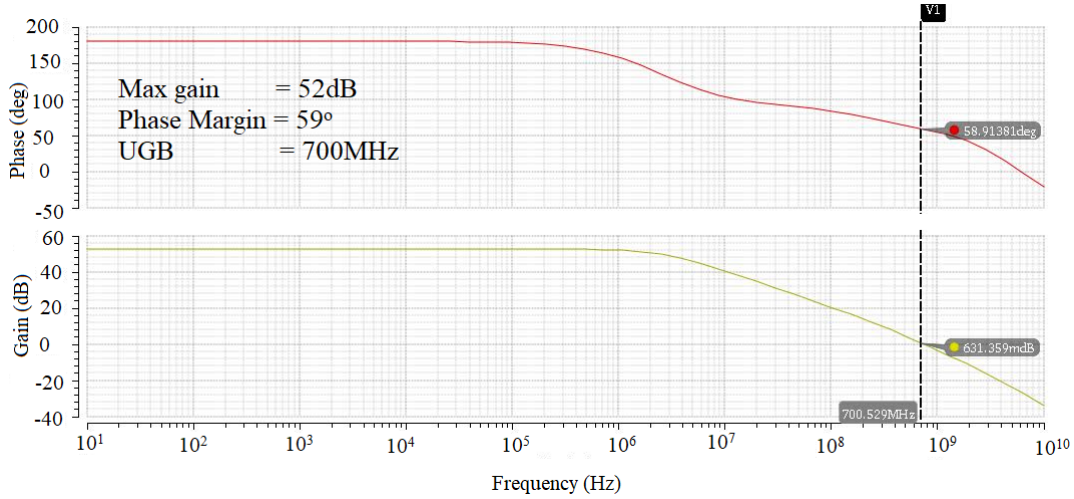

Figure FA1.6: Low gain op-amp Bode plots (with a 1.2pF load)

FA1.7 shows the output spectrum (N=2048) of a sample and hold amplifier for 1.55MHz and 1.91MHz full scale analog sine input which is sampled at 4 MSPS by the op-amp.

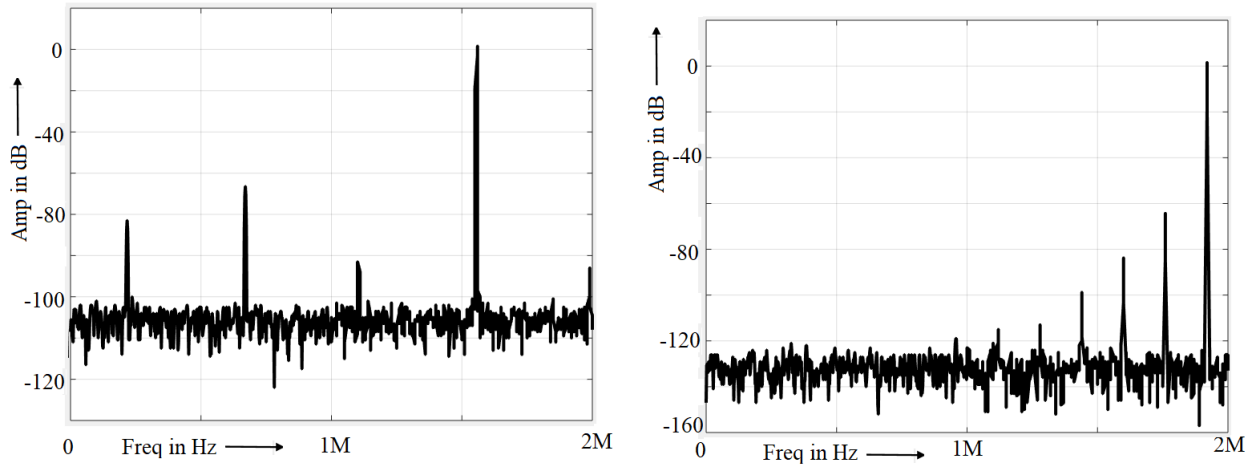

Figure FA1.7: Output spectrum of the low gain op-amp (with a 1pF load)

A.3 Bootstrap Switch

FA1.8 shows the circuit of the bootstrap switch. It operates with a single clock and turns ON and OFF the switch (M1) with respect to clock (clk). With *clk* low, the transistors M9 and M10 discharges the gate the M1. At the same time, VDD is applied onto the capacitor C0 by turning on M3 and M8. During the ON phase, C0 acts as a battery connected to V_{GS} of M1 to turn ON the switch. Transistors M2 and M7 are used to isolate the capacitor from the switch while the capacitor is charging. When *clk* goes high, M3 pulls down the gate of M7, allowing charge from the capacitor on the gate of M1. Hence both M1 and M2 are turned on. M2 enables the gate of M1 to track the input voltage V_{in} shifted by VDD, keeping the gate-source voltage constant independent of the input signal. Actual switch is highlighted in FA1.8.

The transient response of the differential bootstrap switch for sine wave input is as shown in FA1.9. FA1.10 shows the output spectrum (N=2048) of a 0.45MHz and 1.94MHz full scale analog sine input which is sampled at 4 MSPS by the bootstrap switch.

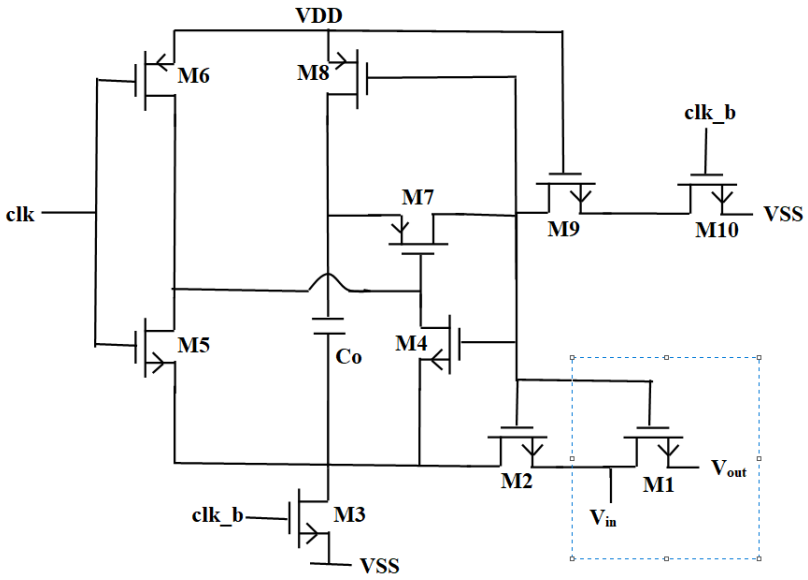

Figure FA1.8: Bootstrap Switch

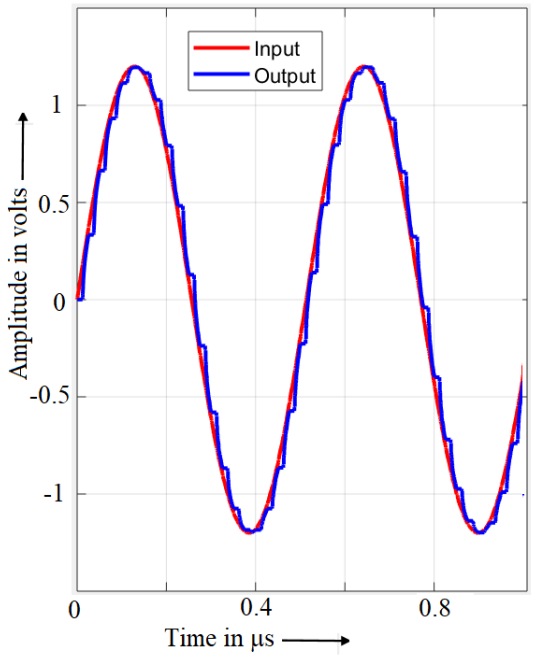

Figure FA1.9: Transient response of Bootstrap Switch

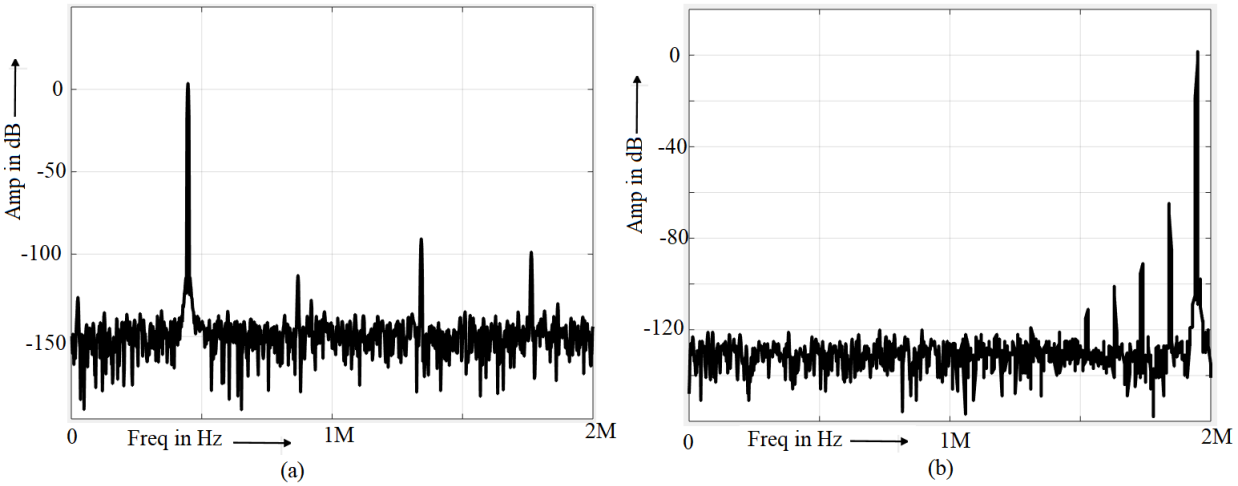

Figure FA1.10: Output spectrum of Bootstrap Switch (a) 0.45MHz and (b) 1.94MHz

A.4 StrongARM Latch

The StrongARM latches are widely used in sense amplifiers, comparators etc. The circuit of the latch is as shown in FA1.11. The circuit provides rail-to-rail outputs at X_B and Y_B in response to the polarities of V_{in+} and V_{in-}. The latch consists of a clocked differential pair represented by M1-M2, two cross-coupled pairs, represented by M3-M4 and M5-M6. There are four precharge switches, M9, M10, M11, M12. In the first phase, when *clk* is low; M1 and M2 are off; the nodes X, Y, P and Q are precharged to VDD. In the second phase, when *clk* goes high, M9, M10, M11, M12 turn off. Input transistors M1 and M2 turn on, drawing a differential current that is proportion to V_{in+} and V_{in-} . With M3-M6 off, the differential current allows the difference between V_p and V_q ($|V_p - V_q|$) to grow and possibly extend beyond $|V_{in+} - V_{in-}|$. This is the third phase, called amplification phase since it provides voltage gain. As V_p and V_q changes, the cross-coupled NMOS transistors (M3-M4) turn on. This allows the drain currents of M1 and M2 to flow from X and Y. The positive feedback around these transistors eventually brings one output back to VDD while allowing the other to fall to VSS. If V_{in+} is greater V_{in-}, the the output X_B eventually raises to VDD and Y_B falls to VSS

and vice versa.

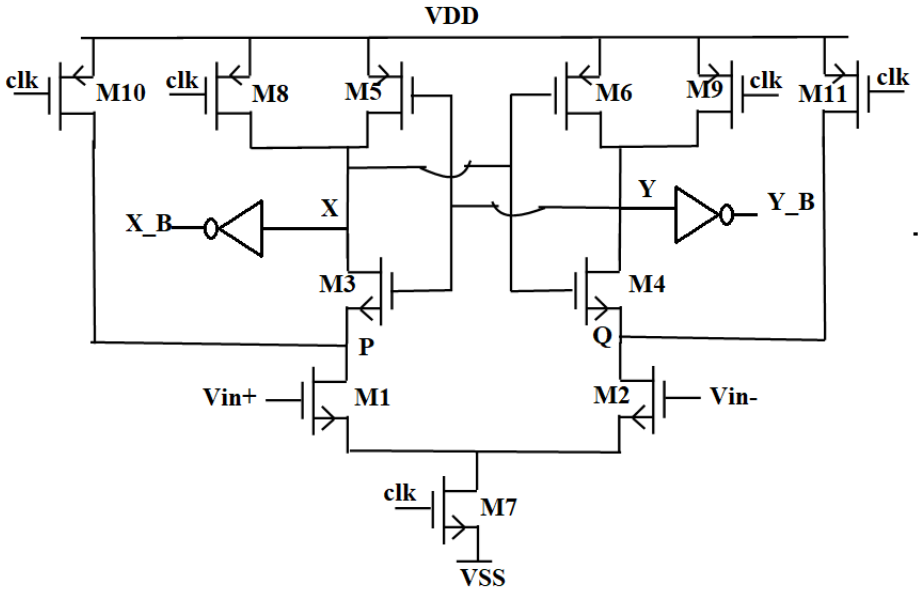

Figure FA1.11: StrongARM latch

A.5 Switched Capacitor Differential Comparator

The sub ADC in the pipeline stage consists of 2 fully differential comparators. It is required to compare the differential input $V_{in+} - V_{in-}$ against the thresholds $\frac{V_{ref}}{4}$ and -$\frac{V_{ref}}{4}$ for an ADC input ranging between V_{ref} and -V_{ref} to obtain the digital output code of that particular stage. The switched capacitor comparator shown in FA1.12, operates in 2 phases using 2 non overlapping clock signals, $clk1$ and $clk2$. During the phase when $clk2$ is high, the reference voltage V_{ref} is sampled onto the capacitor C, while the capacitor 3C is shorted to ground. During the phase when $clk1$ is high, the differential input voltage is applied on to both the capacitors 3C and C resulting in a voltage proportional to $V_{in} - \frac{V_{ref}}{4}$. If this voltage is greater than zero, then the output of the comparator, built using StrongARM latch, B1, is high and vice versa. This is equivalent to comparing V_{in} with -$\frac{V_{ref}}{4}$.

Another comparator is required to compare V_{in} with $\frac{V_{ref}}{4}$. The voltages V_{ref} and -V_{ref} in the circuit shown in FA1.12 are swapped in order to obtain a voltage proportional to $V_{in} + \frac{V_{ref}}{4}$. If this voltage is greater than zero, then the output of the comparator (built using StrongARM latch), B0, is high and vice versa.

The output digital code of the ADC is calculated based on B1 and B0 as shown in TA1.1.

Table TA1.1: Output digital codes of 1.5 bits/stage

B1	B0	Output Digital code
0	0	-1
0	1	0
1	1	+1

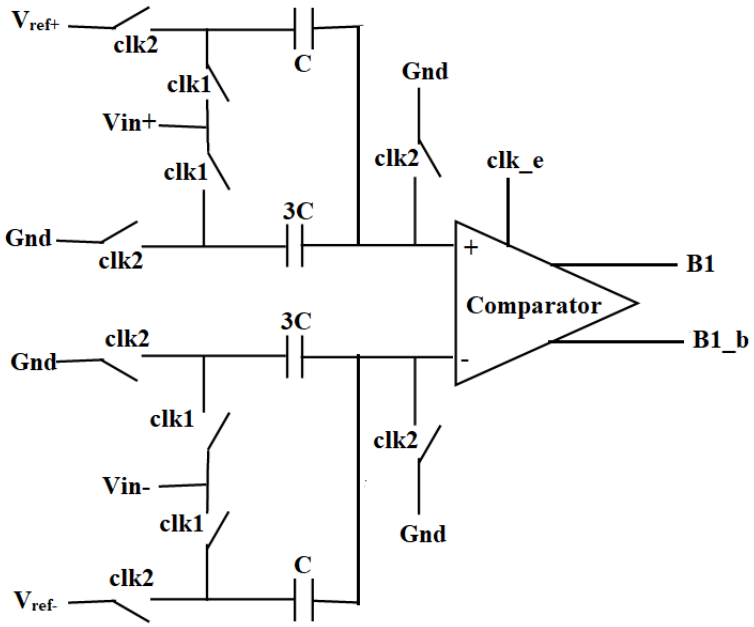

Figure FA1.12: Switched Capacitor Differential Comparator

A differential input signal varying from -1.2V to 1.2V is given as the input to both the comparators. The output of the comparators B1 and B0 are latched. FA1.13 shows how the outputs B1 and B0 vary as the differential input varies from -1.2V to 1.2V. B1 = 0 and B0 = 0 for a differential input less than -300mV. B0 changes to 1 for a differential

input between -300mV and 300mV while B1 stays at 0. For input greater than 300mV, B0 = 0 and B1= 1.

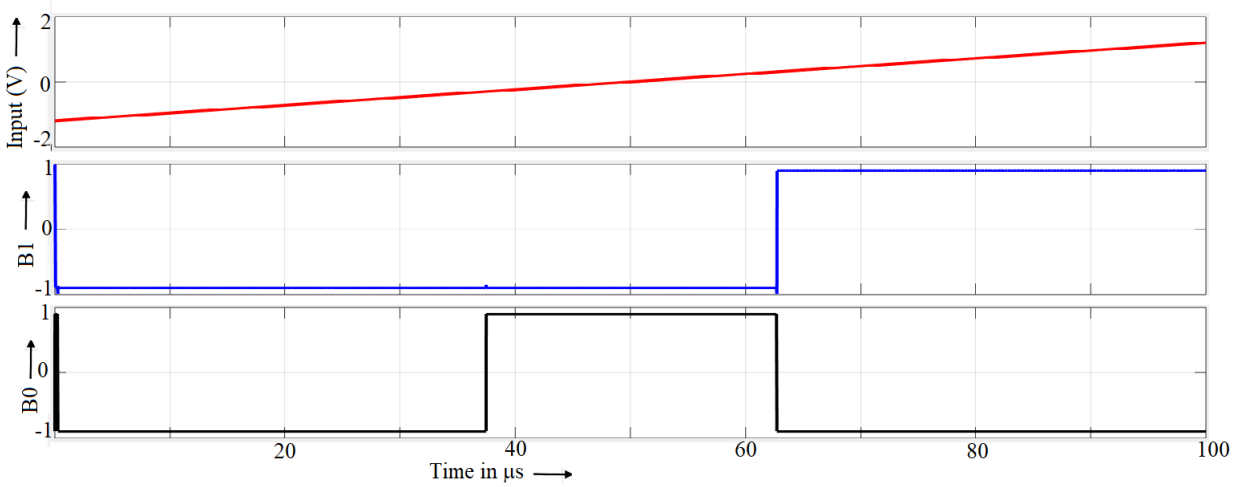

Figure FA1.13: Output of Differential Comparator

A.6 Two Bit Flash ADC

The 2 bit Flash ADC forms the last stage of the pipeline ADC. This consists of 3 comparators built using Strong ARM latch. The resistors used in Flash ADC are connected as in *Resistor String DAC*. Since the input varies from $-V_{ref}$ to $+V_{ref}$, one end of the resistor string is connected to $+V_{ref}$ and the other end is connected to $-V_{ref}$. The output across each resistance is compared with the analog input, V_{in}, using a series of comparators. The output of the comparator is high, if V_{in} is greater than the voltage across the resistance to which the other input of the comparator is connected. The output of the comparators which is in the form of a thermometric code (T_0, T_1, T_2)is then converted to a binary code (D_0, D_1)by a digital encoder. Since the comparison operation can be done in parallel, it just takes one clock cycle for data conversion. The block level representation of the 2 bit Flash ADC is as shown in FA1.14. FA1.14 shows the single ended version of the comparator while a differential version is used in the

actual circuit.

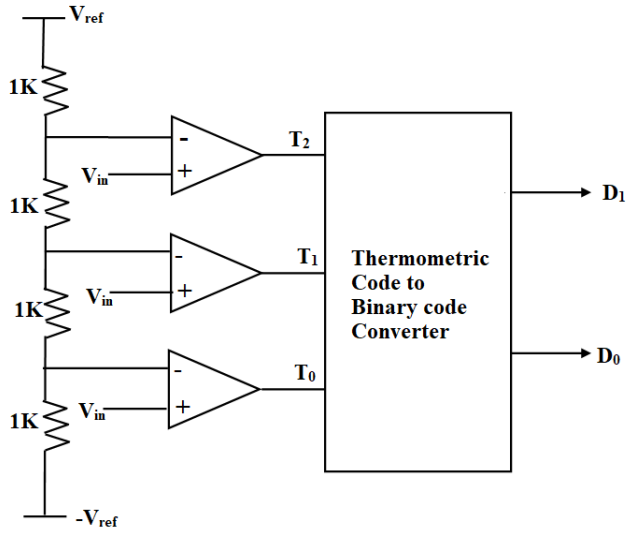

Figure FA1.14: 2 bits Flash ADC

The output of the 2 bits flash ADC is as shown in FA1.15. B1 = 0 and B0 = 0 for a differential input less than -600mV. B1 = 0 and B0 = 1 for an input that lies between -600mV and 0. B1 = 1 and B0 = 0 for an input that lies between 0 and 600mV. B1 = 1 and B0 = 1 for an input greater than 600mV.

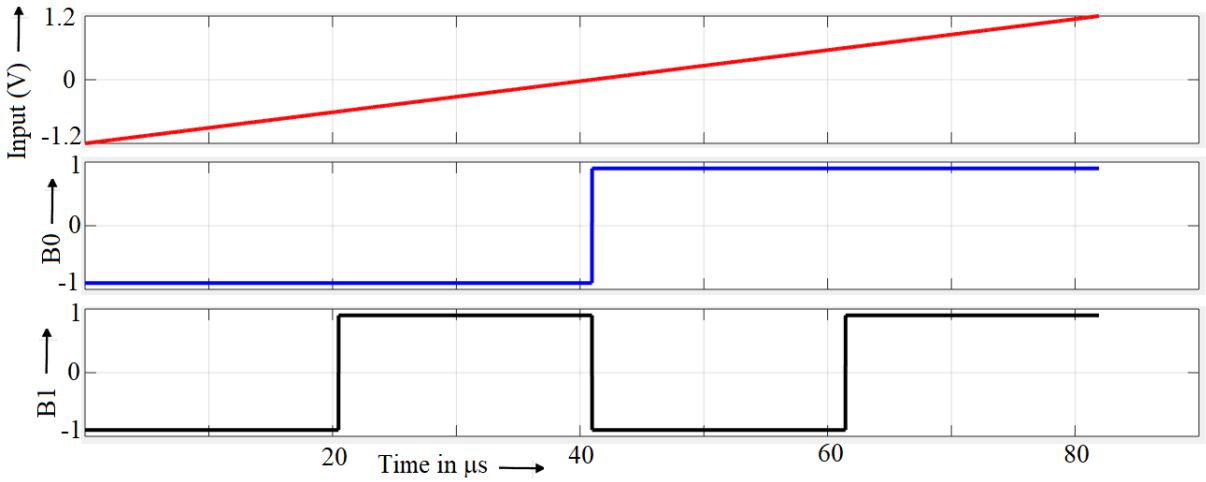

Figure FA1.15: Output of 2 bits Flash ADC