World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Advancing precision in pipeline analog-to-digital converters: Leveraging MATLAB for design and analysis in next-generation communication systems

Joseph Nnaemeka Chukwunweike [1,*], Chinonso Emeh [2], QS Husseini Musa Kehinde [3] and Caleb Kadiri [4]

[1] *Automation and Process Control Engineer, Gist Limited, Bristol, United Kingdom.*
[2] *Automation Engineer, University of South Wales, United Kingdom.*
[3] *Quantity Surveyor, Robert Gordon University, United Kingdom.*
[4] *Embedded System and Software Analyst, Morgan State University, Baltimore, Md, United States.*

World Journal of Advanced Research and Reviews, 2024, 23(01), 2333–2383

Publication history: Received on 08 June 2024; revised on 20 July 2024; accepted on 22 July 2024

Article DOI: https://doi.org/10.30574/wjarr.2024.23.1.2172

Abstract

The integration of MATLAB in designing and analyzing pipeline Analog-to-Digital Converters (ADCs) addresses the complexities of next-generation communication systems. This study reviews literature to identify trends and challenges in high-speed ADC designs. Mastery of MATLAB and its toolboxes, like Simulink and Signal Processing, is essential for advanced ADC modeling and analysis. Key challenges include maintaining accuracy with diminishing supply voltages in modern semiconductors. The study explores incorporating high-voltage analog components and develops strategies for superior accuracy using low-voltage transistors. MATLAB and pipeline architecture are crucial for advancing ADC design, providing innovative solutions for next-gen communication systems.

Keywords: MATLAB; Pipeline ADC's; High-speed ADC design; Low supply voltage challenges; ADC accuracy; Next-generation communication systems

1. Introduction

The history of Analog-to-Digital Converters (ADCs) is intertwined with electronic and digital technology advancements. ADCs convert continuous analog signals into digital data, essential for modern electronics. Early ADC concepts emerged in the late 19th century, tied to telegraphy, telephony, and electrical measurement. Samuel Morse's 1830s telegraph and Joseph Henry's telegraph register laid the groundwork for digital communication. The Wheatstone Bridge, introduced by Sir Charles Wheatstone in the 1840s, was crucial for measuring electrical resistance.

In 1927, Harry Nyquist's sampling theorem established the foundation for modern ADCs. ADC development accelerated in the 1950s and 1960s with vacuum tubes and discrete components, and the advent of integrated circuits in the 1970s and 1980s revolutionized ADC design. Successive Approximation Register (SAR) ADCs and Flash ADCs improved speed and accuracy. The 1980s introduced pipeline ADCs, achieving high speed and resolution. Sigma-delta ADCs, popular in the late 20th century, used oversampling for high resolution, ideal for audio and precision measurements.

High-performance applications, like broadband communication systems, demand ADCs with high resolution and bandwidth. Pipeline ADCs are favored for their efficiency and speed. However, low-voltage digital processes present challenges, increasing costs with traditional designs. This thesis explores these challenges using MATLAB for analysis and proposes innovative methodologies to achieve high performance in low-voltage processes, focusing on pipeline ADCs.

* Corresponding author: Joseph Nnaemeka Chukwunweike

ADCs facilitate analog-to-digital interactions across diverse applications, from digital radio to medical sensors. ADC architecture varies by resolution, speed, power consumption, and latency, necessitating careful selection based on application demands. MATLAB aids in this process, particularly for pipeline ADCs. The rise of digital communication and technologies like DSL highlights the growing need for high-performance ADCs, driven by consumer demands for greater bandwidth for applications like video streaming (Siragusa E, Galton I, 2004).

1.1 Concepts

Analog-to-digital converters (ADCs) are pivotal in bridging analog and digital domains across diverse applications, including digital radio, military and medical sensors, and communications. Each ADC architecture offers unique benefits in resolution, speed, power consumption, and latency, necessitating careful selection based on application requirements, often utilizing MATLAB for optimization, particularly for pipeline ADCs.

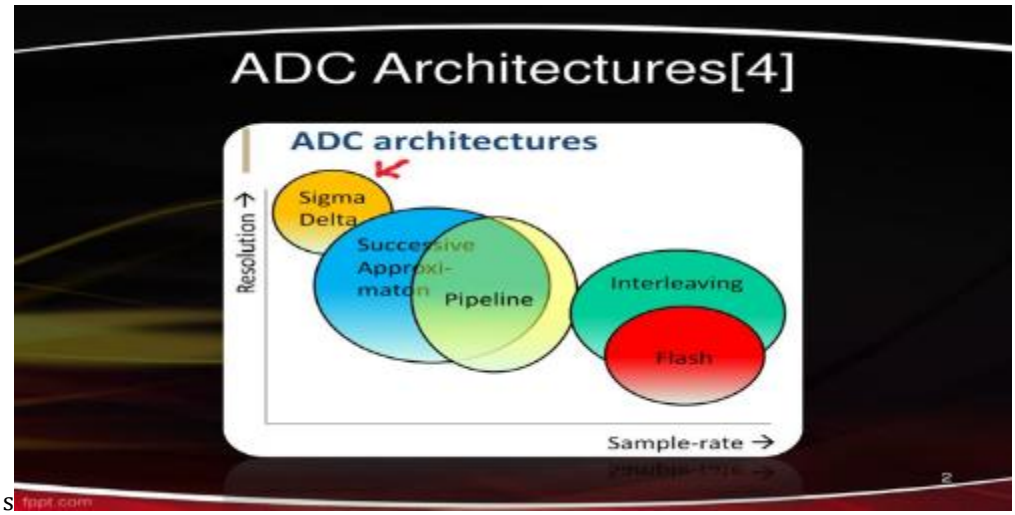
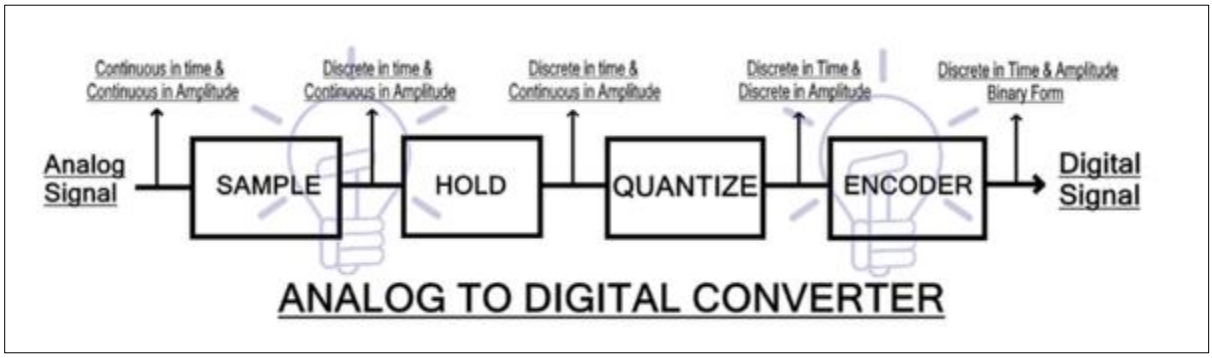


Figure 1 ADC Architecture



, Figure 2 Analogue-to-Digital Converter

The widespread adoption of digital communication technology, combined with advancements in wired and wireless communication, has increased the demand for high-performance Analog-to-Digital Converters (ADCs). This is especially evident in the escalating speed requirements for DSL (Digital Subscriber Line) systems, which utilize the previously untapped bandwidth of existing twisted-pair telephone lines originally used for low-bandwidth voice data. DSL technology leverages this unused bandwidth to enable high-speed digital communications. Current standards for DSL variants like Asymmetric DSL (ADSL) and Very high-speed DSL (VDSL) require ADCs to meet conversion rates of 2.5 MS/s and 24 MS/s, respectively, with resolutions typically between 13 to 14 bits . As consumer demand for greater bandwidth grows, driven by applications such as video streaming, these requirements are expected to increase.

To address these needs, research has focused on two primary domains: the improvement of Nyquist-rate ADCs and innovations in oversampling ΔΣ ADCs, as shown by the arrows in Figure 2.

1.1.1 Amelioration of Nyquist-Rate ADCs

Pipeline ADCs, in particular, are the focus of much research due to their suitability for applications requiring moderate resolution and high-speed operation. To meet the stringent requirements of digital communication systems, pipeline ADCs need enhancements in accuracy. Recent studies in modern analog processes have achieved effective resolutions of 12 bits , while earlier designs using older analog processes reached 13 bits . These improvements are critical for maintaining the performance standards demanded by current and future communication technologies.

1.1.2 Innovations in Oversampling ΔΣ ADCs

Oversampling ΔΣ ADCs are preferred for applications requiring high to very-high resolution at low to medium speeds. These ADCs achieve high resolution by oversampling, which trades bandwidth for increased resolution through the averaging of multiple samples. Noise shaping techniques using z-domain functions further enhance the performance of ΔΣ ADCs. Recent efforts have focused on improving the speed of these ADCs while maintaining their high resolution. This balance is crucial for applications that need precise data conversion without compromising on speed.

The demands of digital communication systems and other applications have driven continuous advancements in ADC technology. For instance, in the context of DSL, the growing consumer need for higher bandwidth to support applications like video streaming necessitates ADCs that can handle higher data rates and resolutions. This, in turn, fuels ongoing research and development in both Nyquist-rate and oversampling ADCs to meet and exceed these evolving requirements. As technology progresses, we can expect further innovations in ADC design and implementation, ensuring that they continue to play a vital role in the advancement of digital communication and other fields.

Objective of study

The main objectives of this dissertation are to:

- Review current literature to understand high-speed ADC design trends and challenges in next-gen communication systems.
- Master MATLAB and relevant toolboxes for robust ADC modeling, simulation, and analysis.
- Investigate the impact of diminishing supply voltages on ADC design, and trade-offs with accuracy in emerging communication applications.
- Explore high-voltage analog components to mitigate low supply voltage constraints, considering manufacturing costs and strategies to achieve high ADC accuracy using low-voltage digital transistors.
- Enhance the precision of pipeline ADCs, utilizing MATLAB and pipeline architecture for advanced ADC design, simulation, and analysis, highlighting their role in advancing research.

Significance of study

This project addresses key research directions in high-speed ADC design amid contemporary semiconductor trends. The shift to lower supply voltages in digital CMOS processes enhances speed and efficiency but poses challenges for ADC accuracy, which relies on input signal power. While high-voltage analogue components can mitigate these constraints, they increase manufacturing costs for mixed-signal SoCs. This research focuses on enhancing the precision of pipeline ADCs and minimizing costs, utilizing MATLAB and pipeline architecture as crucial tools. MATLAB and pipeline methodologies significantly advance ADC design, simulation, and analysis, supporting the development of precise, cost-effective ADCs.

2. Literature review

Pipeline analogue-to-digital converters (ADCs) are popular in high-speed, medium-accuracy systems due to their modular structure, which reduces power and area requirements, enabling remarkable speeds. This section covers pipeline ADCs, their common implementations, techniques for performance enhancement, and potential error sources.

2.1 Fundamental Principles of A/D Conversion

Understanding analogue-to-digital conversion (ADC) is crucial. An ADC converts a continuous-time, continuous-value signal into a discrete-time, discrete-value signal through quantization in both time and value domains, often representing voltage. Time-quantization or sampling captures the input signal at discrete intervals. In a basic sampling circuit, closing a switch approximates the input voltage across a capacitor, preserving the signal value when the switch opens.

Signal-level quantization follows, categorized into serial and parallel methods. Serial quantization uses feedback to determine bits sequentially, while parallel quantization, used in Flash ADCs, quantizes the entire signal simultaneously without feedback.

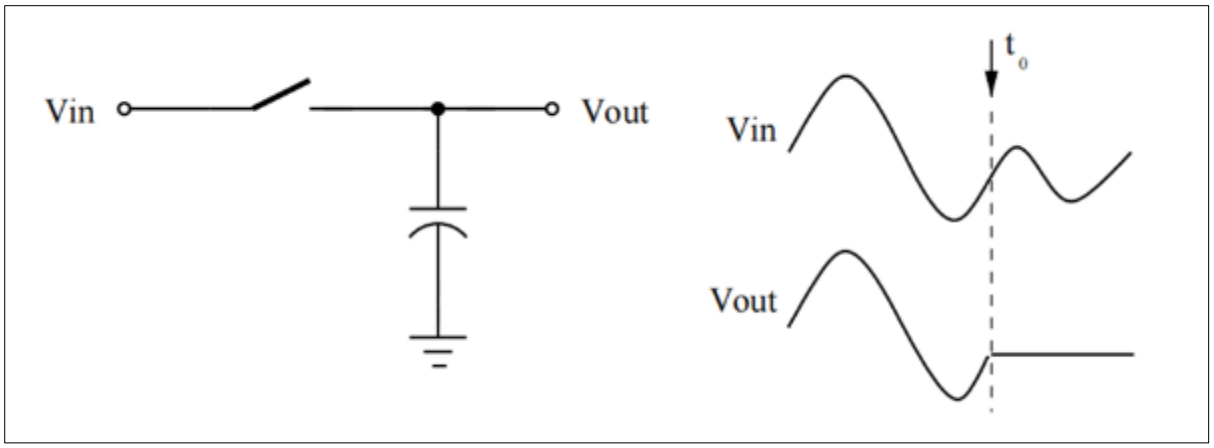


Figure 3 Sampling in A/D Systems

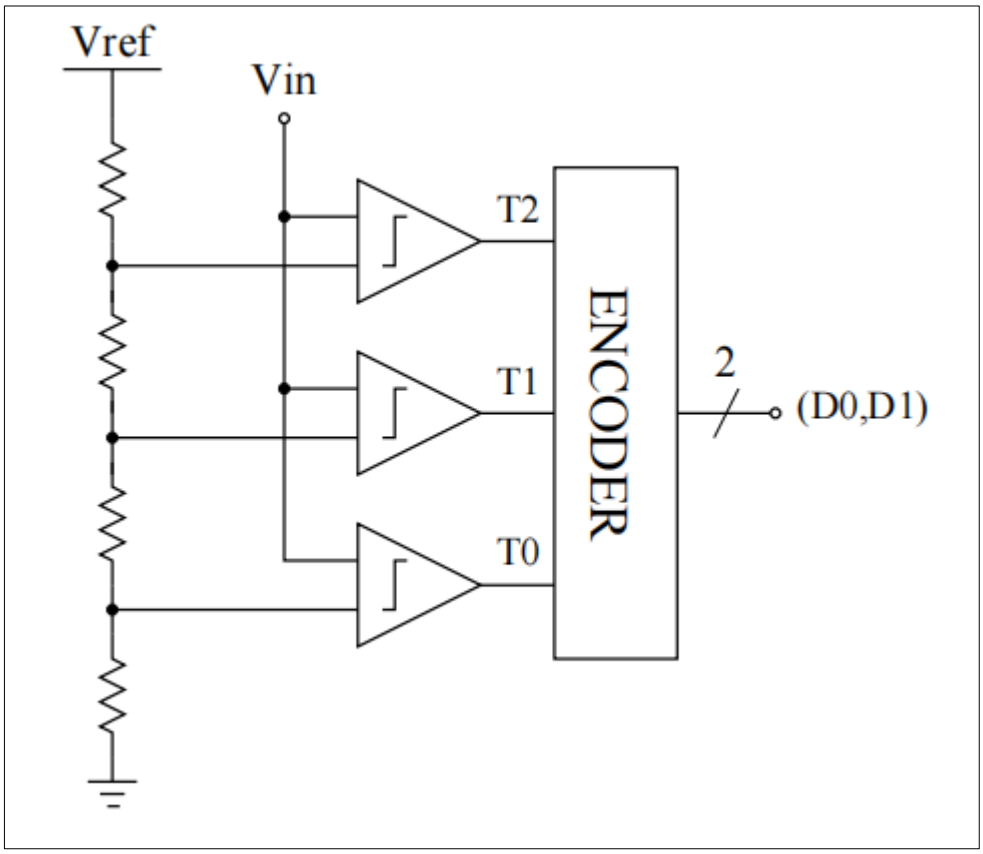


Figure 4 Simple Flash ADC example

The Flash Analog-to-Digital Converter (ADC) performs simultaneous comparisons between the input signal and all possible digital levels, as shown in Figure 4. This concurrent comparison method generates a digital output that approximates the input signal based on an analog reference voltage (VREF). An early example of this concept is found in Peterson's 1979 work.

Figure 2.2 also illustrates an intermediate digital signal known as a Thermometer Code, where the count of ones increases with each value increment, resembling a filling thermometer. For instance, the value '3' in an 8-bit Thermometer Code is '00000111'.

In data converter design, the Least-Significant Bit (LSB) is a critical reference, defined as the converter's full-scale range divided by the total number of quantization levels. For a flash ADC with a 0V-1V input range performing 4-bit conversion, 1 LSB equals 1/16 V. Many ADC specifications, evaluated using MATLAB and pipeline architectures, are expressed in terms of LSBs.

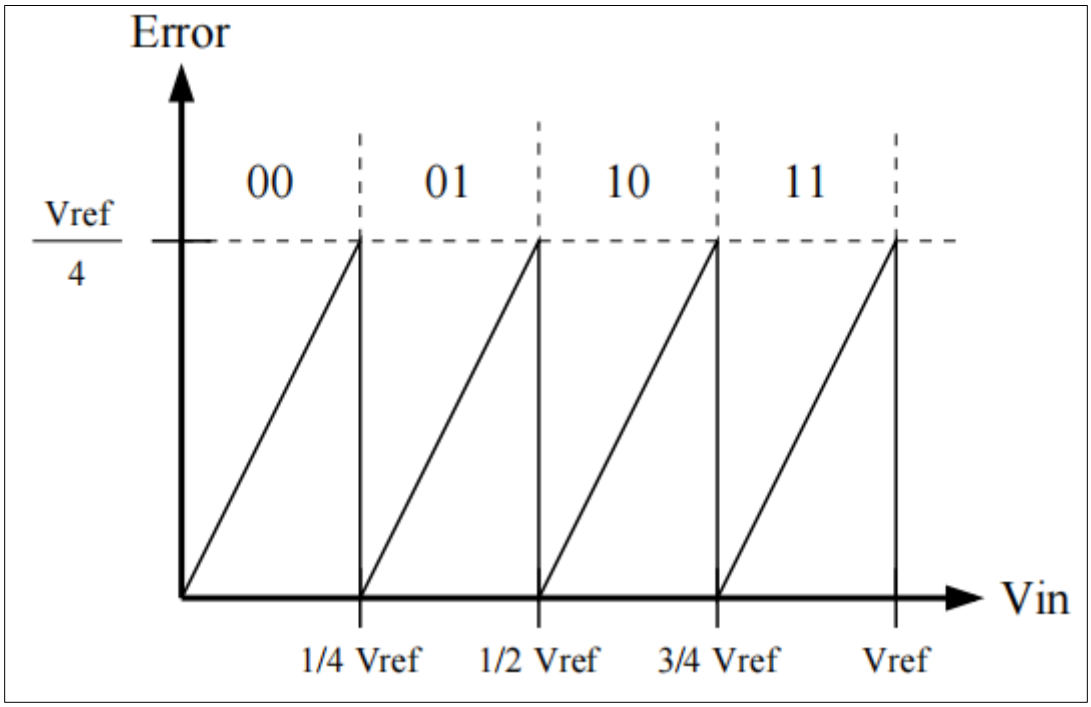


Figure 5 Flash ADC Quantization Error

The quantization step in signal processing pipelines introduces quantization error, as shown in Figure 5. Increasing data conversion precision can reduce this error, but for Flash ADCs, it significantly raises costs. Each additional bit of resolution doubles the required comparisons. Notably, the quantization error contains remnants of the original signal. Extracting and amplifying this error can enable additional conversion steps and higher bit resolution, a concept known as a Two-Step ADC, enhancing high-speed Flash ADC designs.

Quantization error results from subtracting the analog equivalent of resolved digital bits from the input signal, producing a "residue." This residue is amplified to match the analog reference voltage, as depicted in Figure 6.

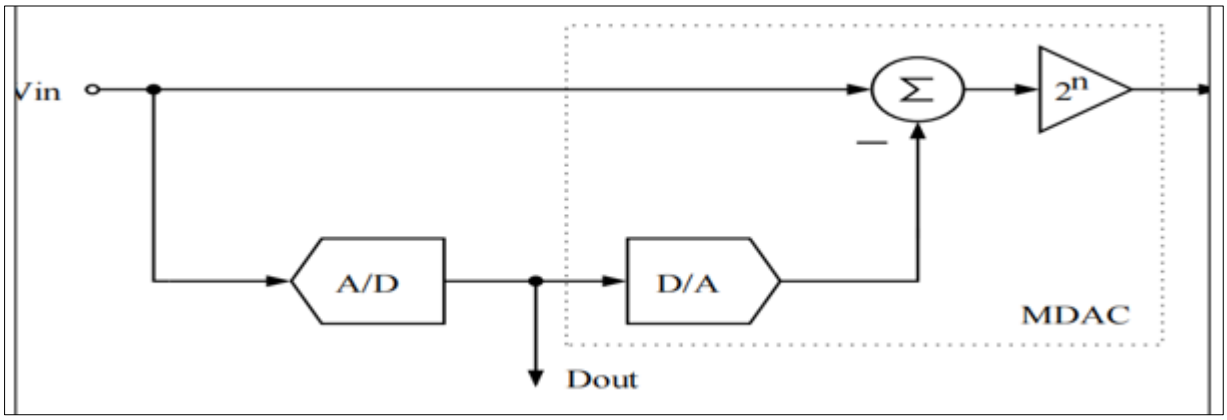


Figure 6 Flash Residue Generation

Digital-to-analog conversion, subtraction, and multiplication are integrated into a Multiplying-Digital-to-Analog Converter (MDAC). The MDAC, prevalent in charge domains, often employs a switched-capacitor topology.

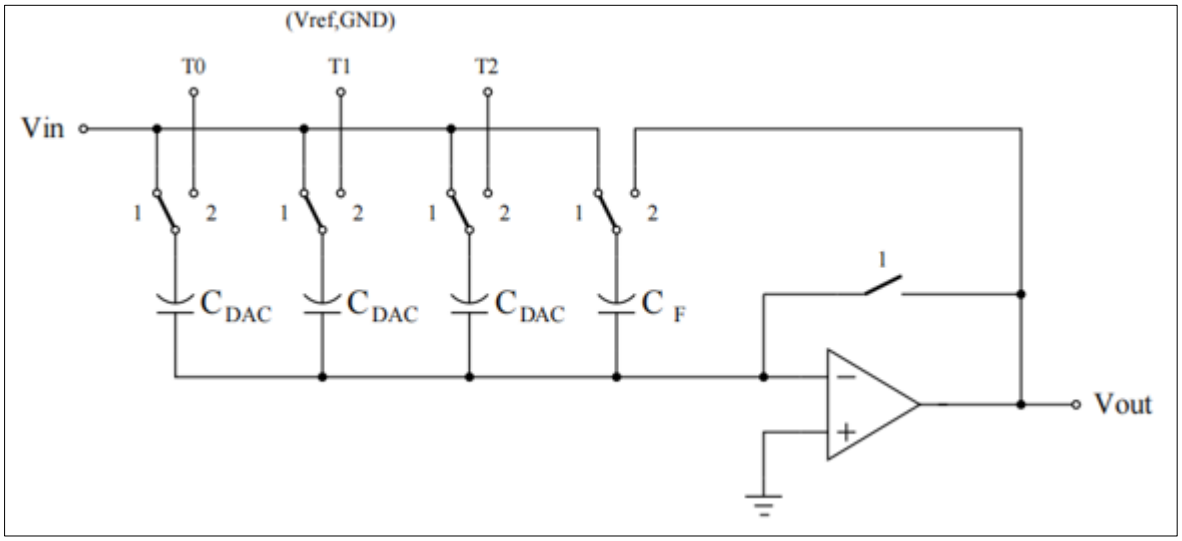


Figure 7 MDAC Circuit Implementation

Figure 7 illustrates this process: during sampling, input charge is deposited onto a parallel combination of ΣCDAC and CF components, then sampled. In the amplification phase, DAC capacitors connect to reference voltages, and a capacitive feedback mechanism transfers charge efficiently to the feedback capacitor (CF).

The number of CDAC capacitors connected to the high reference voltage depends on the digital value from the current stage's quantization process, as shown in Figure 5. The MDAC block's amplification uses a feedback mechanism, where the input value minus the analog representation of the digital input determines the charge transferred to CF, with voltage amplification calculated using \(Q = CV' \).

2.2 Enhancing resolution with MATLAB: the pipeline ADC

The Two-Step ADC concept can be expanded with additional MDAC and comparator stages, forming the basis of a sophisticated design known as the pipelined ADC. Figure 5 represents a single stage of a Pipeline ADC, which is replicated to achieve the desired output bits, as shown in Figure 8. Comprehensive texts, such as Johns and Martin (1997), detail pipelined ADC intricacies. Typically, a Sample-and-Hold (S/H) stage is introduced at the pipeline's front end to aid the first stage's conversion.

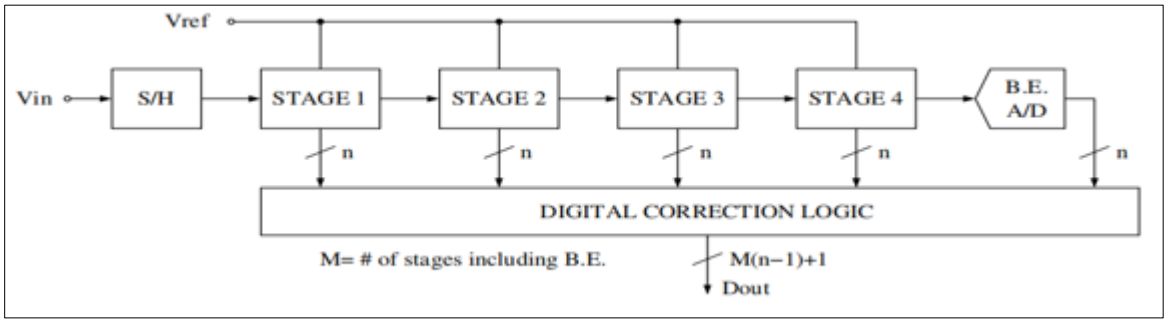


Figure 8 Pipeline ADC

2.3 Per-stage resolution in pipeline ADCS

The pipeline ADC architecture in MATLAB offers designers flexibility in selecting the quantizer resolution for each stage. A common implementation is the 1.5 bits-per-stage configuration, which provides faster conversion times and reduced power consumption by lowering demands on the sub-ADC component. This is particularly effective for lower to medium resolution ADC designs.

In contrast, Multi-Bit Pipeline ADCs use higher sub-ADC resolutions, beneficial for high overall ADC resolutions. The main advantage of multi-bit stages is the significant amplification at the first MDAC stage, which reduces noise from subsequent stages, crucial for high-resolution designs. This increased gain allows for aggressive reduction in size and power consumption of back-end stages. The term "Back-End A/D" (B.E. A/D) refers to the final stage, typically a flash ADC. This scaling strategy ensures more power and area for the critical initial stages, enhancing pipeline ADC design in MATLAB.

2.4 Digital redundancy in pipeline ADCS

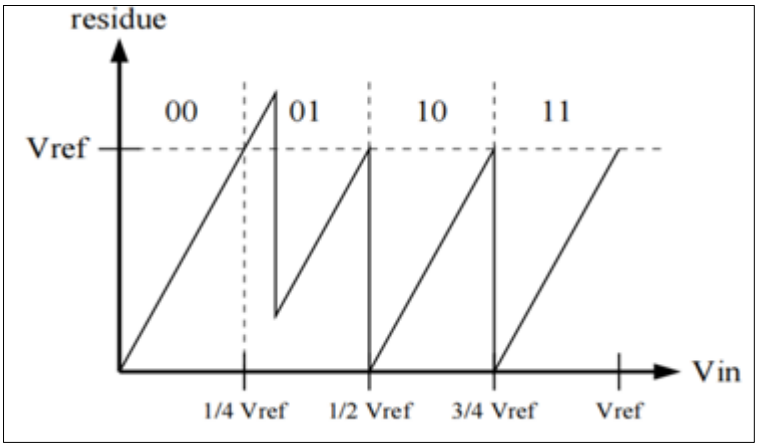


Figure 9 Residue Error Sample

In pipeline ADCs, selecting sub-ADC resolution must consider the comparator block's precision, crucial for the overall ADC system's resolution and linearity. Ignoring this can cause comparison errors that saturate the lower bits. MATLAB is essential for addressing and analyzing these issues in pipeline ADC design.

Imagine the vivid scene depicted in Fig. 9 within the dynamic world of a pipeline ADC. During the '00' to '01' transition, a comparison error emerges, causing the residue voltage to exceed the ADC reference voltage boundaries. This anomaly leads to subsequent stages producing numerous 'ones,' introducing pronounced non-linearities in the converted signal—a landscape of chaos unfolds. Conversely, errors in the opposite direction result in all subsequent conversions becoming 'zeros,' contrasting with a sense of stillness.

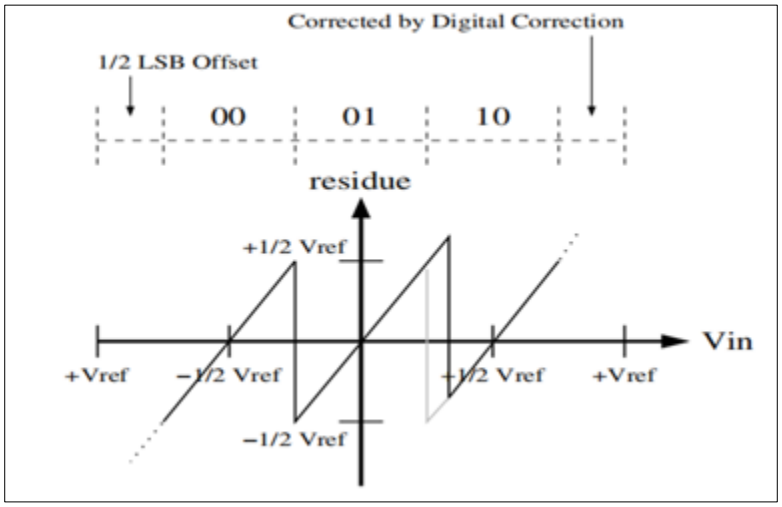


Figure 10 Digital Correction Residue

However, the challenge extends beyond saturation errors. Initial stage comparison errors significantly impact the ADC's overall accuracy. The first stage comparator must maintain precision aligned with the entire ADC's requirements, a daunting task due to inherent offset levels from imperfect reference generation and comparator mismatches.

Techniques like offset-storage (Razavi & Wooley, 1992) offer potential solutions but increase circuit complexity. Despite these efforts, saturation problems persist, posing ongoing challenges in pipeline ADC design.

In the vibrant realm of a pipeline ADC illustrated in Fig. 9, a '00' to '01' transition unveils a critical comparison error, causing the residue voltage to surpass ADC reference boundaries. This discrepancy triggers subsequent stages to generate numerous 'ones,' inducing significant non-linearities in the converted signal—a chaotic landscape unfolds. Conversely, errors in the opposite direction force all subsequent conversions to 'zeros,' painting a contrasting scene of stillness.

Beyond saturation issues, initial stage comparison errors profoundly affect ADC accuracy. The first stage comparator must uphold precision that meets the ADC's stringent requirements, challenging due to inherent offsets from imperfect reference generation and comparator inconsistencies. Techniques like offset storage (Razavi & Wooley, 1992) offer promise but escalate circuit complexity. Despite these efforts, saturation challenges persist, continuing to pose significant hurdles in pipeline ADC design.

2.5 Data converter distortions

In the realm of data converters, imperfections manifest as Differential Non-Linearity (DNL) and Integral Non-Linearity (INL), quantified in LSBs. DNL measures sequential code divergence from the ideal 1 LSB, ideally at 0 LSB. INL aggregates DNL values across all codes. DNL exceeding 1.0 LSB can cause missing codes; limiting DNL to < 0.5 LSB mitigates this risk (Fig. 11). Multi-stage ADCs like Pipeline ADCs face additional challenges, such as non-monotonic behaviors due to gain variations between stages. MATLAB is instrumental for evaluating and mitigating these issues in pipeline ADC design, particularly for systems with integrated feedback loops.

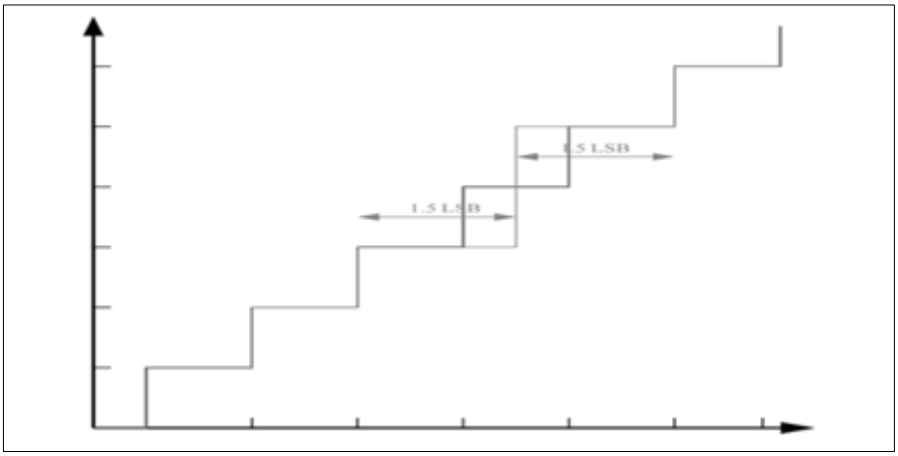


Figure 11 Missing Code DNL Condition

Assessing linearity in ADCs includes Differential Nonlinearity (DNL) and Integral Nonlinearity (INL). Dynamic evaluation using single-tone sine waves reveals harmonic content, complementing static measurements for comprehensive ADC performance assessment.

2.6 Pipeline error origins

Pipeline ADCs, renowned for their efficiency in power and speed, encounter unique challenges in error propagation due to their segmented design.

2.6.1 Offset Error

Offset errors, simple additive errors, introduce a constant error into the signal. Unlike traditional ADCs where rectifying offset errors directly impacts the output, mid-pipeline offsets in pipeline ADCs can lead to significant nonlinearities. Managing these errors requires specialized design techniques to minimize signal offset:

- *Charge injection*: Inadvertent charge addition during MOSFET switch deactivation is a common origin. Strategies like early-clocking and complementary switch utilization mitigate charge injection offset.
- *Opamp offset*: Compensation methods focus on adjusting opamp offset to minimize its impact.
- *Comparator offset*: Digital correction and offset storage techniques are employed to rectify comparator offset.

- *DAC offset*: Ensuring DAC capacitor matching minimizes output digital code errors.

2.6.2 Gain Error

Gain errors, multiplicative errors affecting the input signal, parallel offset errors in pipeline stages, and can lead to significant nonlinear errors:

- *Feedback capacitor mismatch*: Addressed through design practices like capacitor shuffling.
- *Under-settled discrete-time signals*: Rigorous design and simulation ensure accurate settling within conversion time.

2.6.3 Nonlinear Errors

Pipeline ADCs inherently exhibit certain nonlinear errors:

- *Opamp nonlinearity*: Output impedance variations with gain can be managed by ensuring sufficient open loop gain across the output range.
- *Signal-dependent switch resistance*: Mitigated through complimentary switches and bootstrapping techniques.
- *Parasitic capacitances*: Introduce nonlinearity due to voltage-dependent changes in device characteristics.

Understanding these error origins in pipeline ADCs lays the foundation for enhancing performance and applying effective error mitigation techniques in future designs.

2.7 Noise characteristics

2.7.1 Kt/C Thermal Noise

Kt/C thermal noise arises due to the thermal agitation of charge carriers in resistors and capacitors. In the context of Pipeline ADCs, this noise exhibits a single-pole spectrum limited by the bandwidth of the fundamental switched-capacitor sampling structure (Figure 12). The spectral density of thermal noise at the top-plate capacitor node can be mathematically expressed as:

$$S_{v_{n}}(f) = 4kT(1\C)^2 .1/f$$

where \(k \) is Boltzmann's constant, \(T \) is temperature in Kelvin, \(C \) is capacitance, and \(f \) is frequency.

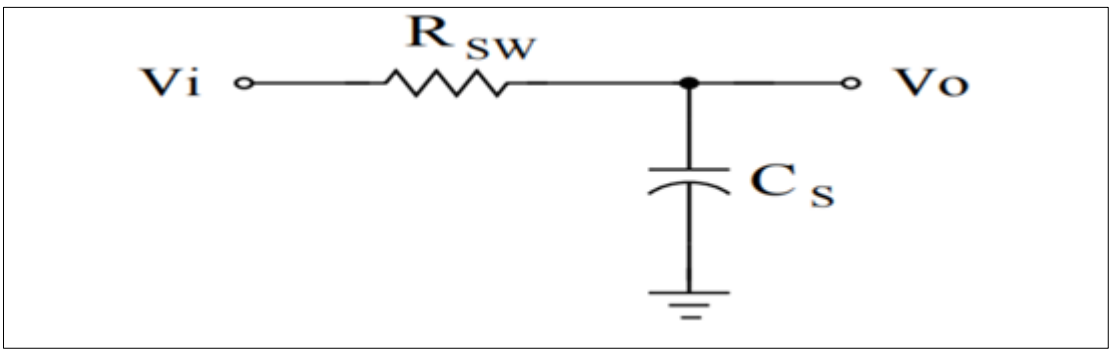


Figure 12 Simple Switched-Capacitor Sample System

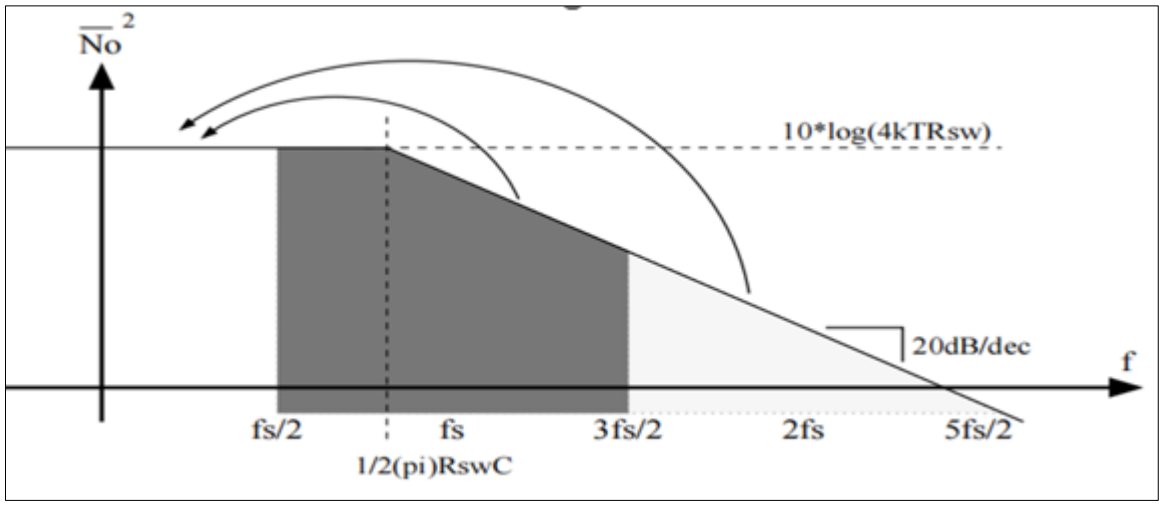


Figure 13 Half-Band kTC Noice Integration

When the switch opens during the sampling process, a variety of frequencies are amalgamated into a frequency range of ±fs/2 (as shown in Fig. 13). Assessing the impact of thermal noise in the sampling circuit involves integrating the noise spectral density across the entire frequency spectrum, from zero to infinity.

$$N_O^2 = \int_0^\infty \frac{4kTR_{SW}}{1+(\omega R_{SW}C_S)^2} d\omega = \frac{kT}{C_S}. \qquad (2.2)$$

Interestingly, in switched-capacitor circuits, the thermal noise characteristics remain unaffected by the properties of the noise source itself. This phenomenon can be explained as follows: as the resistance of the switch increases, the thermal noise originating from the resistor also increases. However, simultaneously, the increased resistance results in a narrowing of the bandwidth of the single-pole system. Consequently, despite the change in resistance, there is no significant alteration in the integrated noise, as depicted in Figure 14

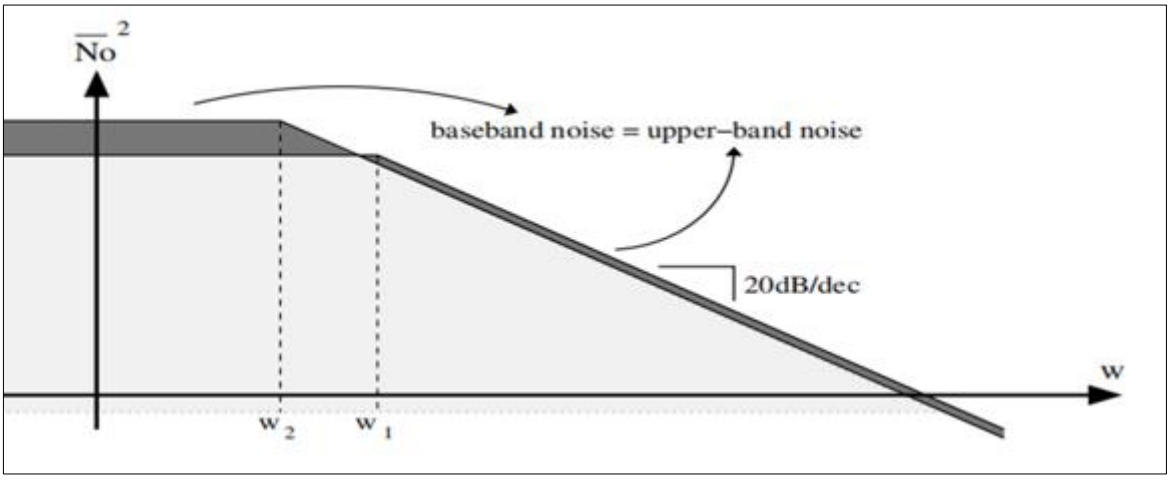


Figure 14 Half-Band Noise Density Variation

Active electronic components contribute significantly to overall noise in systems, typically mitigated by increasing device dimensions and power consumption. However, in switched-capacitor systems, noise is often dominated by kT/C thermal noise and less affected by active component noise, validated through both manual calculations and simulations. The trend towards System-on-a-Chip (SoC) designs integrates analog and digital circuits on a single chip, offering cost and implementation advantages but introducing challenges. Digital components can inject current noise into substrates

and power supplies, potentially affecting analog signal integrity. Strategies to mitigate this issue include timing digital switches to avoid analog processing periods and physically isolating analog and digital sections with separate power supplies and shielding.

2.8 Performance Enhancement

Various circuit and system techniques address these challenges, including offset storage, correlated double-sampling (CDS), and capacitor error averaging. Offset storage, notably introduced by Poujois et al. and standardized by Razavi, is widely used.

2.8.1 Offset Storage

Offset storage involves a two-step process: first, sampling offset voltage onto a capacitor during one clock phase, then integrating it during another phase to subtract from the signal, achieving net-zero offset if offsets align. Razavi's work distinguishes Input Offset Storage (IOS), using a unity-gain feedback network to store pre-amplification offset, crucial in MDACs for pipeline ADCs, and Output Offset Storage (OOS), storing amplified offset at the output for low-power amplifier configurations.

Fig. 15 illustrates these methods, emphasizing their role in enhancing ADC precision and minimizing offset-induced errors in high-speed and high-resolution applications.

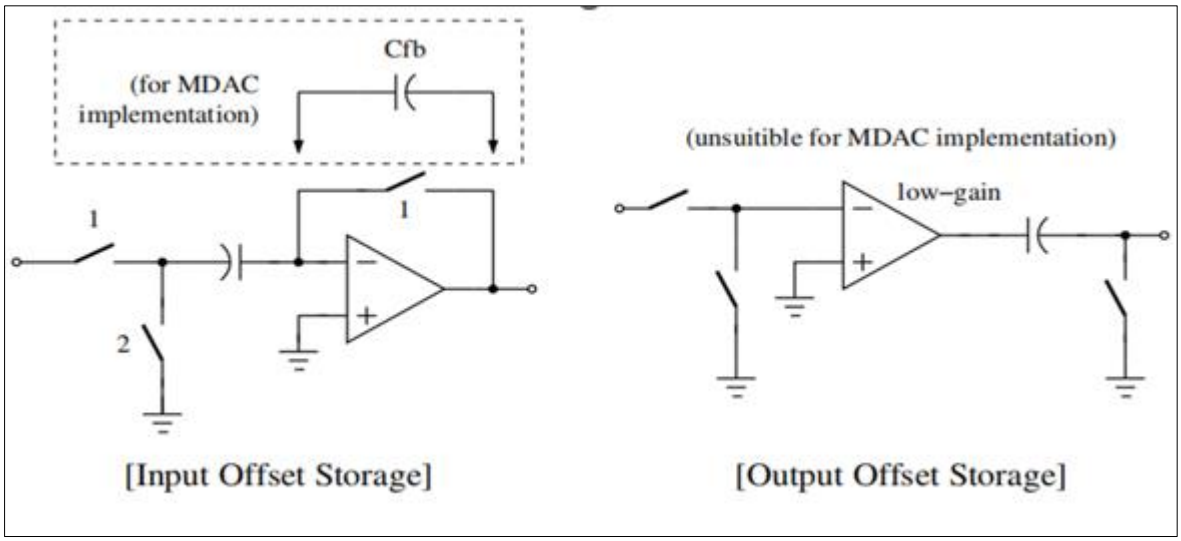


Figure 15 Offset Storage Methods

2.8.2 Correlated Double-Sampling

Correlated Double-Sampling (CDS), a technique outlined by Razavi and Wooley (1992), operates by employing two samples to effectively eliminate correlated signals. This approach shares similarities with offset storage in its noise reduction capabilities but distinguishes itself by significantly enhancing the virtual ground node's efficacy. The method involves sampling the input voltage with two capacitors during the initial phase (CDAC1 + CF B1 and CDAC2 + CF B2). In the subsequent phase, CDAC2 and CF B2 form a feedback loop with the opamp, while another capacitor (CI) captures the value at the virtual ground. This configuration integrates offset and input signal information, enabling precise evaluation of the output signal's error.

Despite its benefits, CDS requires additional time for settling the processed signals, which can be a drawback in applications demanding rapid signal processing. Figure 2.14 illustrates the fundamental operation of CDS, demonstrating its role as a charge-based amplifier essential for minimizing noise and enhancing signal integrity in ADC designs.

$$e_{phase2} = \frac{-1}{A}\left(1 + \frac{C_{DAC1}}{C_{FB1}}\right) V_{o(ph2)}, \qquad (2.3)$$

In this context, Vo represents the output voltage, and A signifies the gain of the open-loop amplifier. Equation (2.3) illustrates a reciprocal relationship between the error and the open-loop amplifier gain. In the concluding phase, when CDAC1 and CF B1 are no longer connected, and CDAC1 and CF B1 are incorporated into the feedback system, the error-storage capacitor CI is introduced into the series with the virtual ground node. This action enhances the virtual ground, resulting in an improved virtual ground. Consequently, the final error observed at the output is approximately inversely proportional to the open-loop amplifier gain.

$$e_{phase3} \approx \frac{-1}{A^2}\left(1 + \frac{C_{DAC1}}{C_{FB1}}\right)\left[\left(1 + \frac{C_{DAC1} + C_I}{C_{FB1}}\right) V_{o(ph2)} - \left(\frac{C_I}{C_{FB1}}\right) V_{o(ph3)}\right]. \qquad (2.4)$$

Equation (2.4) demonstrates that correlated double-sampling results in the amplification of the opamp's effective open-loop gain. The reduced demand for high open-loop gain makes this approach particularly appealing for the design of high-resolution Pipeline ADCs. However, a notable limitation of this method is the additional clock phase needed for the correlated double-sampling process. Recent advancements in this field have led to the development of a technique aimed at eliminating this requirement.

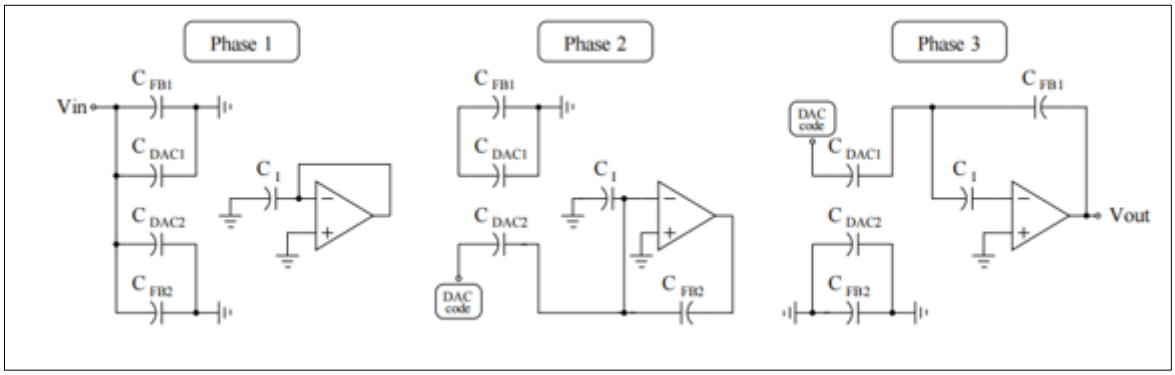


Figure 16 Correlated Double-Sampling

2.9 Capacitor Error Averaging

Capacitor mismatch within Digital-to-Analog Converters (DACs) and charge-based MDAC circuits poses challenges in achieving linearity in Pipeline Analog-to-Digital Converters (ADCs). Historically, achieving precise matching involved costly trimming processes [Mercer, D. (1996)], but modern techniques like capacitor error averaging have emerged to mitigate these issues without the need for trimming.

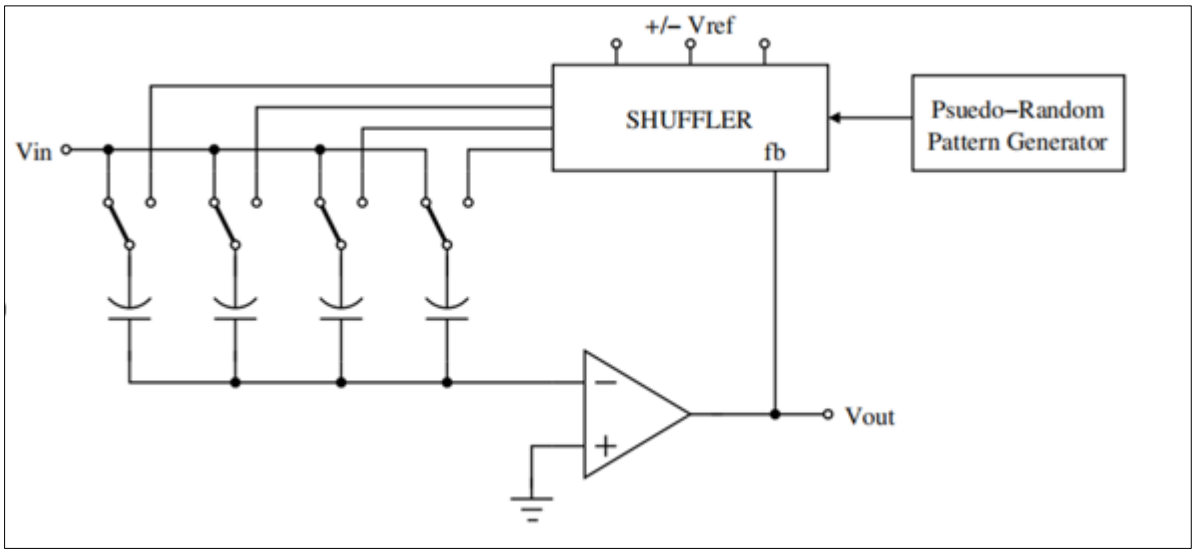


Figure 17 Capacitor Shuffling for DEM

Capacitor error averaging operates by systematically alternating the use of individual capacitors within the MDAC circuit over time. This method effectively averages out mismatches between capacitors, reducing offset and gain errors that lead to non-linearities in ADC performance. Figure 17 illustrates this concept, demonstrating how alternating capacitors can enhance linearity by averaging out device-to-device variations.

Dynamic Element Matching (DEM), initially proposed by Van De Plassche (1976), represents an early form of capacitor error averaging. More advanced techniques, such as Passive Capacitor Error Averaging (PCEA) [Ryu et al., 2004], further refine this approach by averaging errors of opposite polarities, thereby achieving even greater improvements in capacitor matching.

While capacitor error averaging techniques offer significant benefits in improving ADC linearity, they require additional clock phases to facilitate the sequential averaging of multiple capacitor samples. This consideration is crucial when designing high-resolution ADCs aiming for minimal error and optimal performance.

2.10 Calibration Techniques

Calibration serves as a crucial method to rectify inaccuracies in ADCs, offering two primary operational modes: foreground and background calibration.

2.10.1 Foreground Calibration

Foreground calibration involves temporarily halting the conversion process to conduct calibration signal processing. The resultant data from this calibration signal's conversion is utilized to derive calibration information. This information can directly influence output results or adjust ADC operations during regular usage. Foreground calibration, however, may introduce interruptions in normal operation and may require periodic updates.

2.10.2 Background Calibration

In contrast, background calibration operates seamlessly during standard ADC operations. Calibration values are computed concurrently with data conversion. A test signal, often a pseudo-random sequence, is inserted into the real signal, and correlation techniques extract calibration data from the resulting output. Background calibration minimizes disruption to normal operation but may slightly reduce the Signal-to-Noise Ratio (SNR) due to the presence of the test signal.

2.11 Code-Error Calibration

In Pipeline ADCs, mismatched capacitors within MDACs contribute to Differential Non-Linearity (DNL) errors, varying across different digital codes. Code-error calibration addresses this by devising error correction codes specific to each digital code. These correction codes are added to the output code to mitigate inaccuracies associated with each code, thereby enhancing ADC precision.

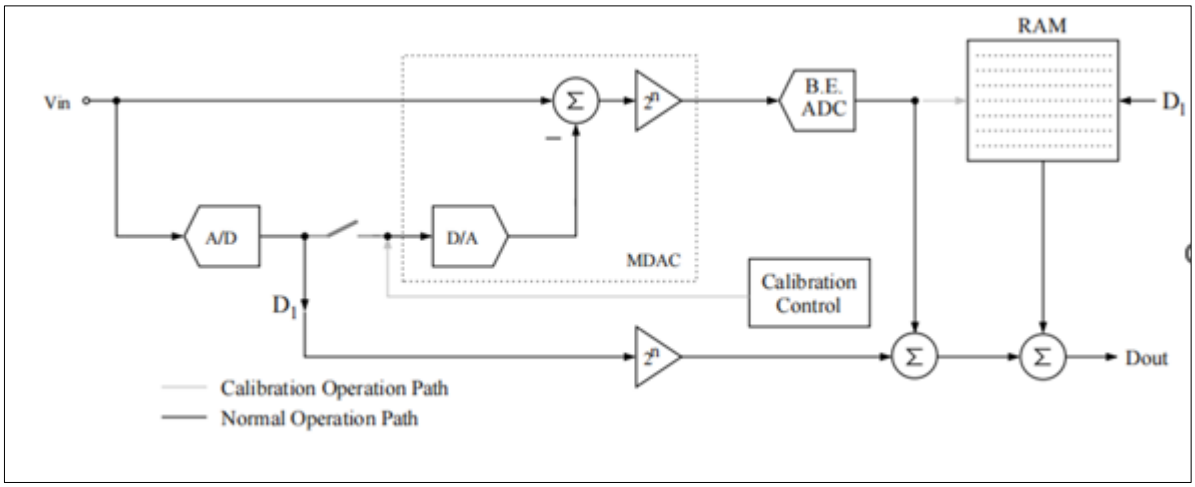


Figure 18 Code-Error Calibration Implementation

In the realm of ADC calibration techniques, the approach begins with quantifying the incremental discrepancy between successive DAC codes. This disparity is crucially evaluated in conjunction with the residual A/D stages within the pipeline. By subtracting the expected value from the actual outcome, the resulting difference represents the error, forming the basis for establishing a corrective code specific to the DAC code under consideration. Figure 18 illustrates this procedural method visually. Importantly, subsequent stages of the Pipeline ADC must exhibit higher resolution than typically required for signal conversion to effectively minimize the impact of thermal noise, often necessitating the averaging of multiple measurements.

The correction codes derived from this process are typically stored in on-chip RAM memory, with addresses corresponding to the primary stage DAC codes. This correction method is straightforward to implement, with the summation during data conversion easily executed in parallel with routine processing tasks.

Another notable calibration technique, Radix Calibration, addresses potential gain errors arising from capacitor mismatches, finite MDAC amplifier gain, and linear settling inaccuracies in Pipeline ADCs. In standard 1.5 bit-per-stage Pipeline ADCs, where each stage has a nominal radix of two, the expected base value for each stage's resolved code aligns with this radix of two. If a revised radix value is applied in the ADC system, an "ideal digital code" (radix two code) can be generated through simple calculations using individual radix values associated with each stage's code conversion in the pipeline, as depicted in Figure 19. This approach provides a comprehensive solution to mitigate sources of error in Pipeline ADCs, ensuring accurate and reliable performance across varying operating conditions.

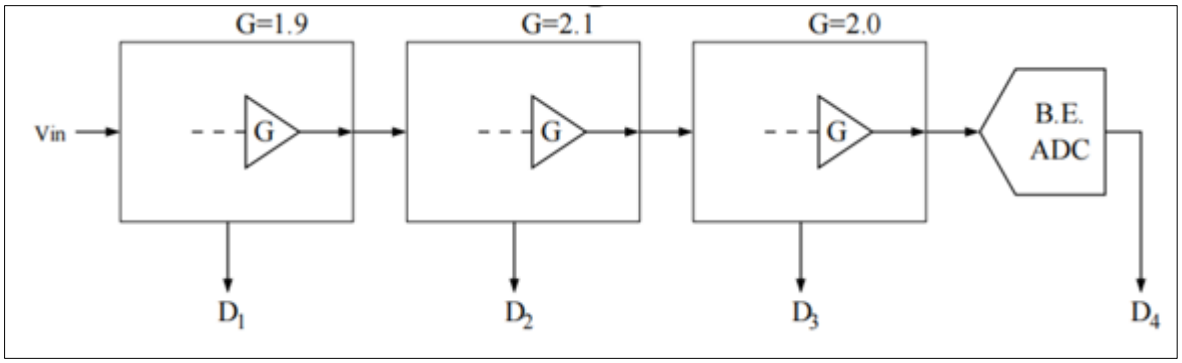


Figure 19 Radix Calibration

$$decimal\_value = (1.9)(2.1)(2)D_1 + (2.1)(2)D_2 + (2)D_3 + D_4. \qquad (2.5)$$

This approach presents an elegant solution tailored for 1.5 bit-per-stage Pipeline ADCs. However, it's worth noting that its implementation can be computationally demanding. Additionally, there are substantial processing requirements in terms of conversion time due to the necessity of floating-point multiplications for digital output correction. In the realm of non-linear calibration, the focus is on mitigating non-linear errors introduced by components exhibiting non-linear behaviour within individual pipeline stages. Key sources of non-linearity often include MOSFET switches and operational amplifiers. This calibration technique allows for the utilization of low-power and low-complexity analogue circuits within the ADC, albeit at the expense of increased digital complexity required to rectify errors arising from these circuits. Recent instances of this calibration approach can be found in references [Murmann, B., & Boser, B. (2003).] The generation of radix values for each code conversion at every stage in the pipeline (as illustrated in Figure 2.17) can be achieved through straightforward calculations.

3. The design and methodology

This section focuses on the critical design considerations for high-resolution, medium-speed Pipeline ADCs aiming for 14 bits of Effective Number of Bits (ENOB) at a sampling rate of 20 Mega-Samples Per Second (MSPS). Achieving such high resolution places stringent demands on the initial stages of the pipeline, necessitating careful selection of the system architecture to optimize power consumption and silicon area utilization.

The role of noise, particularly kT/C sampled thermal noise, is pivotal in enhancing the Signal-to-Noise Ratio (SNR) by a single bit. To achieve the targeted 14-bit ENOB, a significant increase in sampling capacitance—approximately fourfold—is required. Given that recent ADC designs typically achieve around 12 bits of ENOB, the aspiration for a 2-bit improvement necessitates a sixteenfold increase in capacitance to elevate SNR performance to the desired level. Balancing this SNR enhancement with cost considerations underscores the need for ingenuity and dedication at the system-level design phase.

3.1 Bit-Per-Stage Architecture

In Pipeline ADC design, a fundamental architectural decision revolves around the number of bits resolved per stage. Lower resolutions per stage favor higher speed but lower overall resolution ADCs. Conversely, higher resolutions per stage, termed multi-bit resolutions, are more suited to applications demanding stringent accuracy, as outlined in this project.

3.1.1 Trade-offs in MDAC Design

The Multiplying Digital-to-Analog Converter (MDAC) design for the Pipeline ADC is influenced by three critical metrics: conversion speed, converter accuracy, and noise. These metrics are fixed design requirements that impact the associated costs such as power consumption and silicon area utilization. Figure 20 provides a simplified illustration of the MDAC structure for reference throughout this discussion.

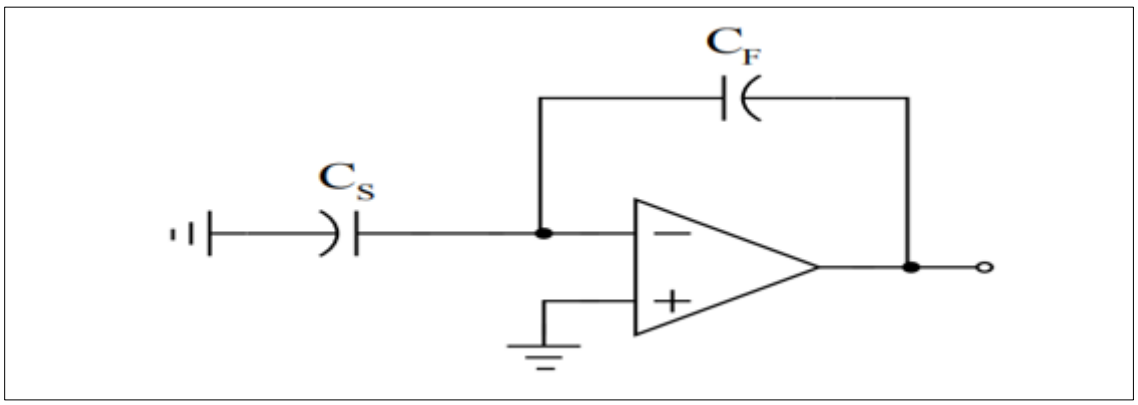


Figure 20 Simple MDAC Reference (Amplification Phase)

3.2 Velocity of conversion and its intricate connection to per-stage resolution

The transformation speed of an ADC is intricately linked to the resolution of each individual stage, primarily governed by the loop feedback factor. This factor, β, crucially determines the amplification of each stage, which in turn impacts the overall performance of the ADC. The feedback factor β is defined by the ratio of the feedback capacitor CF to the sum

of the sampling capacitor CS and the feedback capacitor CF, as shown in Equation (3.1). This relationship directly affects the gain G of the stage, which is 2^n, where 'n' denotes the number of bits resolved per stage. Substituting the stage gain into the equation for β gives us Equation (3.2), where β = 1 / (G + 1). A key initial design parameter is the loop unity-gain bandwidth (UGBW), which, in conjunction with capacitive loads and the feedback factor β, determines aspects such as opamp power consumption and device sizes. The relationship between the loop UGBW and the feedback factor β is described by Equation (3.3), highlighting how amplifying the stage gain increases the bandwidth requirements, thereby affecting power consumption.

The precision of the ADC is critical for achieving its overall accuracy requirement, such as a 12-bit accuracy. This necessitates that each stage in the pipeline, especially the sample-and-hold stage, processes the analog signal with precision exceeding the target bit accuracy to maintain fidelity through subsequent stages. In high-speed data conversion systems, strategies like load scaling between stages can optimize power efficiency needed for settling, but these approaches become less effective as conversion speeds approach technological limits. Hence, single-bit-per-stage or 1.5-bit-per-stage architectures are common choices for balancing speed and power consumption in ADC designs. Overall, managing the feedback factor β, loop UGBW, and stage resolution is crucial for optimizing power consumption while ensuring the required precision and accuracy in Pipeline ADC designs.

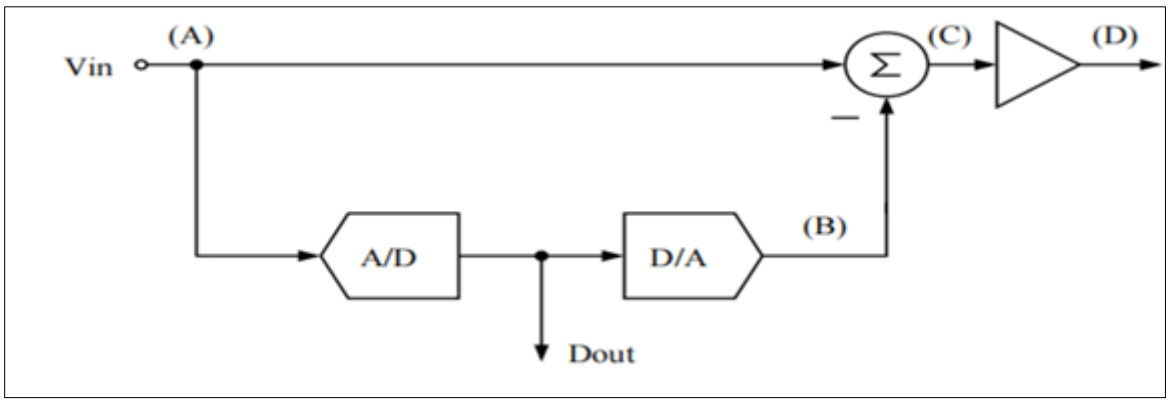


Figure 21 System-Level Pipeline ADC Stage

It seems there might be a misunderstanding or some errors in the equations and descriptions provided. Let's correct and clarify the concepts step by step.

3.3 Equation Corrections and Clarifications

3.3.1 Equation (3.4)

The provided equation seems to be misaligned and not fully clear. Typically, the loop unity-gain bandwidth (ωt) for an ADC stage is related to its settling time (t_settle) and the desired accuracy (M). The correct expression should reflect how the settling time impacts the bandwidth requirement, possibly in a logarithmic relationship. However, the provided equation structure doesn't clearly convey this. Here's a revised attempt to express the relationship:

\[

\omega_t = \frac{-\ln(2^{-(M-n)})}{t_{\text{settle}}}

\]

This equation suggests that as the resolution (n) of each stage increases, the settling time (and thus the bandwidth requirement) decreases logarithmically to maintain accuracy.

3.3.2 Equation (3.5) and (3.6)

These equations seem to describe the noise contributions in the ADC. However, they are not entirely clear in their current form. Typically, for sampled noise in MDACs, the noise density is proportional to the capacitance and temperature (kT/C noise). Let's clarify:

- *Equation (3.5)* should ideally express the output-referred noise due to thermal noise and charge injection in the MDAC. It might look like:

\[

v_{\text{noise,out}} = \sqrt{4kT \left(\frac{C_S}{2} + C_F \right)}

\]

This equation accounts for the thermal noise contributions from both capacitors involved in the sampling process.

- *Equation (3.6*) needs revision for clarity, but it seems to attempt to relate the noise to the feedback factor and gain. A correct expression should account for the noise characteristics of each component in the MDAC.

3.4 Sampled Noise and Design Considerations

3.4.1 Impact of Stage Resolution (n):

Increasing the resolution per stage in a Pipeline ADC reduces the required closed-loop bandwidth (ωt), which helps mitigate power consumption. This reduction in bandwidth requirement stems from the slower dynamics required for higher resolution stages.

3.4.2 Sampled Noise and System Architecture:

In high-accuracy ADC designs, managing noise contributors like thermal noise (kT/C noise) and ensuring proper system architecture (such as balanced MDAC designs) are crucial. The equations aim to quantify these noise sources relative to the stage resolution and feedback factors.

$$\overline{v}_{n(in)}^2 = \frac{2kT}{C_S}\left(\frac{G\,(G+1)}{G^2}\right) = \frac{2kT}{C_S}\left(\frac{G+1}{G}\right). \qquad (3.8)$$

Actual MDAC implementations would typically employ the feedback capacitor CF for sampling as well, reducing the G + 1 term to G. In this case, Equation (3.8) demonstrates that sampled, input-referred noise is independent of per-stage resolution.

3.5 Active Circuit Noise

Active circuit noise can be approached similarly. Figure 22 is employed to formulate the equation describing the dependence of active circuits on per-stage resolution. Two noise sources are shown in the figure, and each source will be independently considered through superposition. The first noise source under consideration is vn1, which models one of the input-pair noise sources. Its contribution to output noise can be calculated as follows:

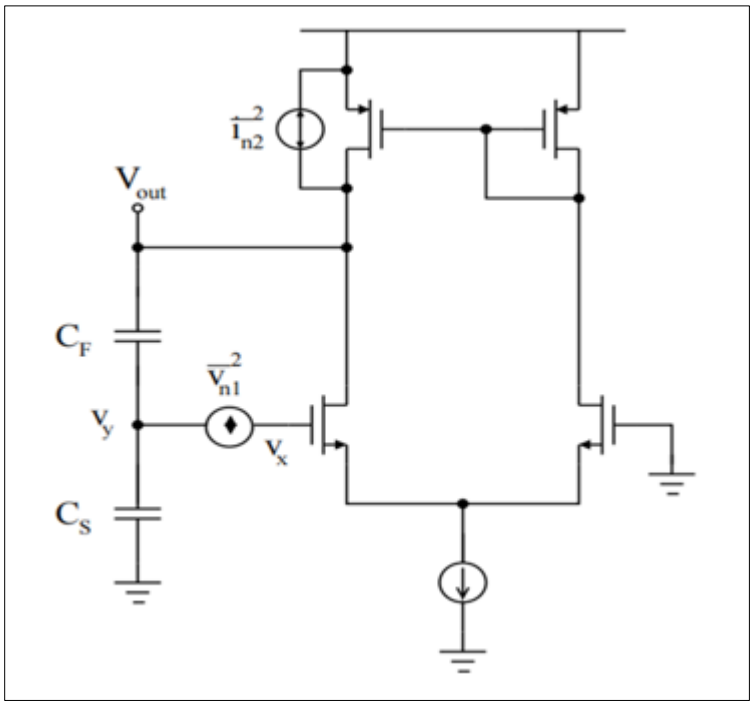


Figure 22 Active Circuit Noise Source Example

Nodes *vx* and *vy* are represented as

$$v_x = -\frac{v_{out}}{A}; v_y = \beta v_{out}, \tag{3.9}$$

In this context, A represents the open-loop gain of the amplifier, and β denotes the feedback-factor established by the capacitive feedback network. The quantity v2n1, signifying the noise source, is expressed as

$$\overline{v}_{n1}^2 = \overline{v}_y^2 - \overline{v}_x^2 = \beta^2 \overline{v}_{out}^2 + \frac{\overline{v}_{out}^2}{A^2}. \tag{3.10}$$

Rephrasing (3.10) with regard to v2 shows the disclosure of the noise contribution in the output:

$$\overline{v}_{out}^2 = \frac{\overline{v}_{n1}^2 A^2}{1 + \beta^2 A^2} \approx \frac{\overline{v}_{n1}^2}{\beta^2}. \tag{3.11}$$

When we make the assumption that β is approximately equal to 1/G, we can express the input-referred noise as follows:

$$\overline{v}_{in}^2 \approx \overline{v}_{n1}^2. \tag{3.12}$$

This demonstrates an absence of stage resolution dependency concerning input-referred noise.

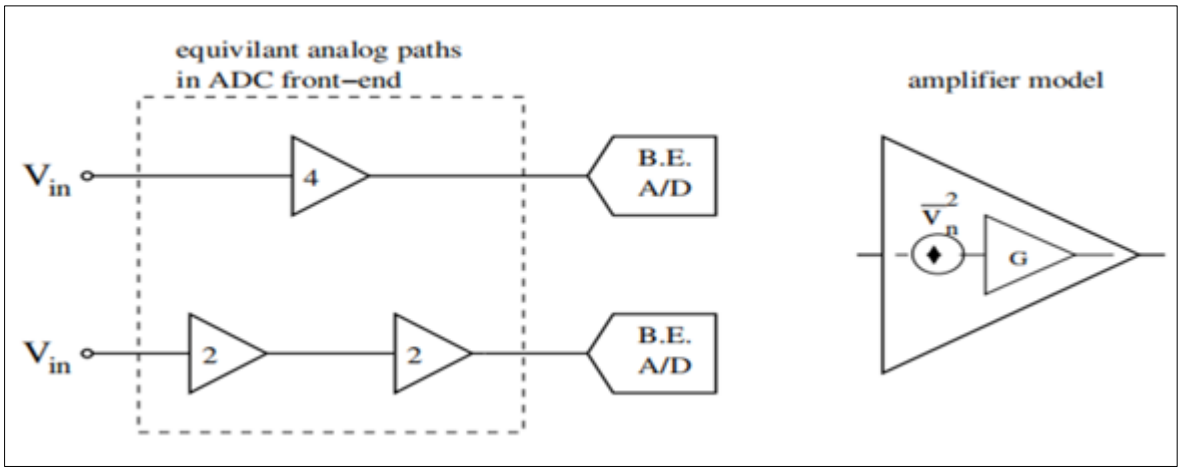


Figure 23 Stage Number Noise Tradeoff Comparison

Next, I turned my attention to the current noise source (i2n2), which signifies a load-related noise source. The noise contribution referred to the output of this source is given by v2out = i2n2ro^2 (Equation 3.13), where 'ro' represents the amplifier's output resistance. Conversely, the same noise contribution, when referred back to the input, is expressed as v2in = i2n2r2oG^2 (Equation 3.14).

It is apparent from Equation 3.14 that there exists a stage-resolution dependence for this noise source. This phenomenon holds true for all amplifier noise sources positioned outside the feedback loop. The implication of Equation 3.14 is that the noise originating from active circuitry can be mitigated (when measured at the input) by employing a higher stage resolution. The number of stages and its relationship with noise also merits consideration. This involves a tradeoff between more stages and fewer stages, which essentially equates to the tradeoff between lower resolution per stage and higher resolution per stage. However, this tradeoff will be explored at a more fundamental architectural level.

In the discussion of high-resolution, medium-speed Pipeline ADCs, the focus is on achieving precise analog-to-digital conversion through careful design considerations and techniques. Here, I'll elaborate on the key points and equations mentioned in your request:

3.6 Differential Non-Linearity (DNL) and Capacitor Matching

3.6.1 Equation (3.15):

Differential Non-Linearity (DNL) in the context of ADCs is crucial for maintaining accurate conversion. The equation:

$$\text{DNL} = k \cdot 2^{(M - n)/2} \sqrt{C_{\text{total}}}$$

explains how DNL improves with increasing stage resolution (n). Here:

- M is the total converter resolution.

- n is the stage resolution.

- k is a parameter related to the capacitor.

- C_{total} is the total capacitance in the DAC.

This equation indicates that increasing the resolution of each stage by one bit reduces the DAC unit-capacitor size by a factor of two, thus reducing DNL by a factor of $\sqrt{2}$.

3.6.2 Capacitor Standard Deviation (Equation 3.16 and 3.17):

The standard deviation \(\sigma_c \) of unit-sized capacitors is crucial for understanding how capacitor mismatches affect DNL:

\[

\sigma_c = k \cdot \sqrt{A_{cu}}

\]

where \(A_{cu} \) is the area of the unit-sized capacitor. This relationship underscores that the variability in capacitor size directly affects DNL.

3.7 Impact on Integral Non-Linearity (INL)

While DNL focuses on the differential error between consecutive codes, Integral Non-Linearity (INL) considers the cumulative error across the entire range of codes. In the context of your simulations (Figures 3.5 to 3.10):

- Simulation Setup: The simulations explore how varying the stage resolution (from 2 to 6 bits in the first stage) affects DNL and INL.

- Capacitor Mismatch Impact: With each increase in stage resolution, the standard deviation of capacitor mismatches worsens by \(\sqrt{2} \), impacting DNL accordingly.

- INL Observation: Unlike DNL, INL does not improve with increased stage resolution because the total error across \(C_{\text{total}} \) remains constant. The error simply gets distributed among more components, reflecting the overall mismatch..

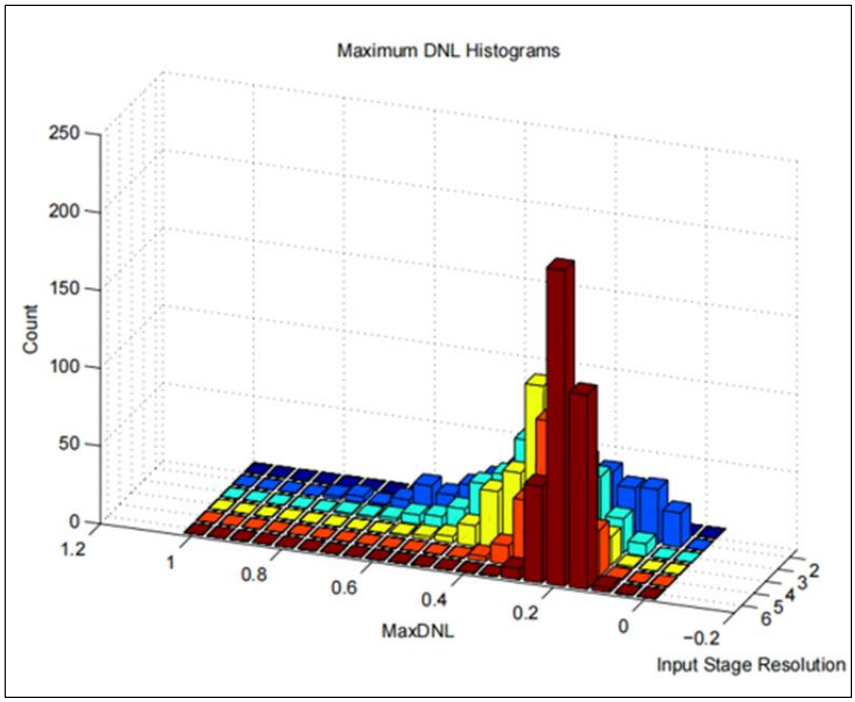


Figure 24 Maximum DNL Histogram Comparison

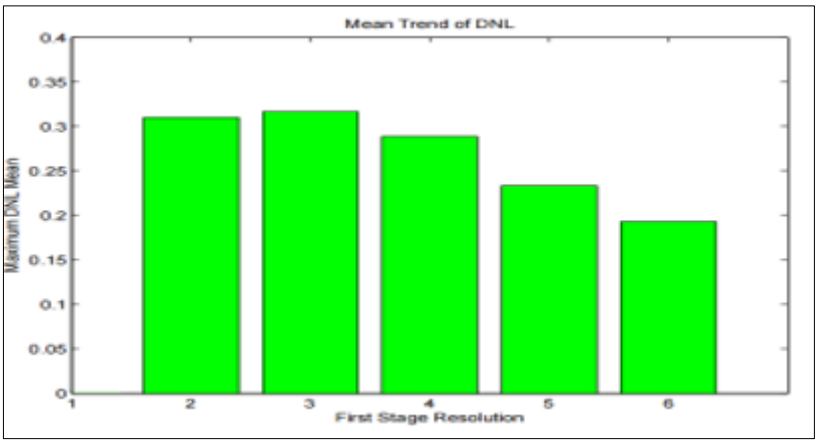


Figure 25 Maximum DNL Mean Over Stage Resolution

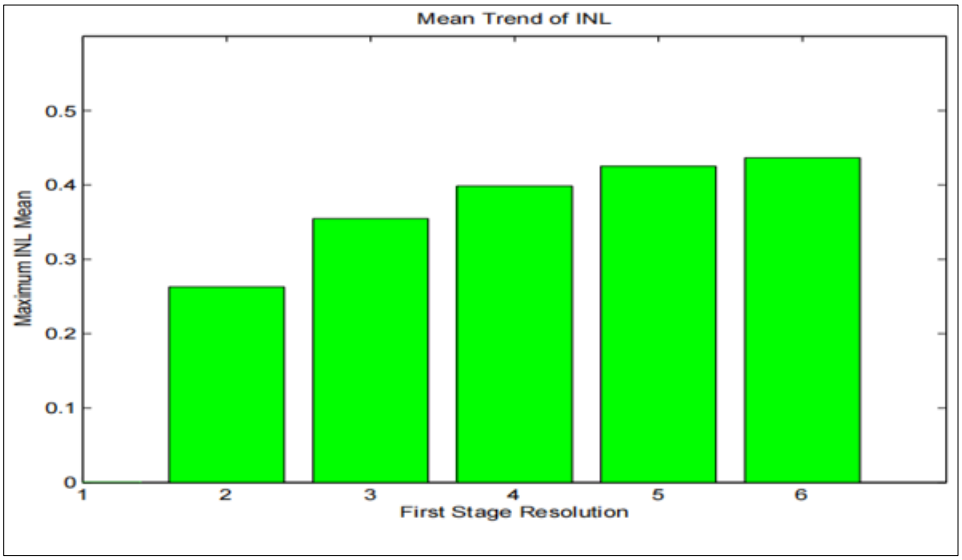


Figure 26 Maximum INL Mean Over Stage Resolution

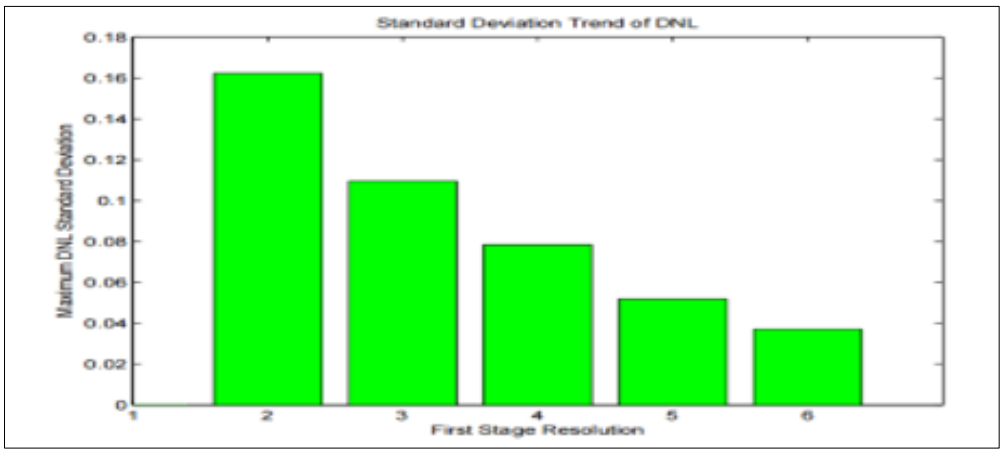


Figure 27 Maximum DNL Standard Deviation Over Stage Resolution

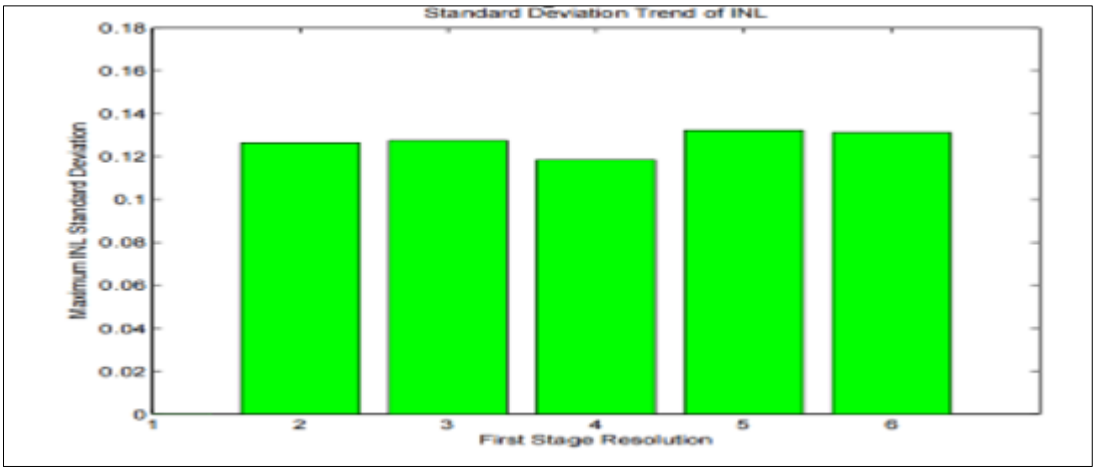


Figure 28 Maximum INL Standard-Deviation Over Stage Resolution

3.7.1 Ponderings Regarding Comparators

In the context of high-resolution Pipeline ADC design, the removal of the Sample-and-Hold (S/H) circuit from the initial stage presents both challenges and benefits:

Benefits of Removing the Sample-and-Hold Circuit

- *Reduction in Power Consumption and Area*: The S/H circuit, which traditionally precedes the first stage of a Pipeline ADC, is responsible for sampling the input signal and holding it constant while the subsequent stages perform their operations. By integrating the sampling function directly into the first pipeline stage, the need for a dedicated S/H circuit is eliminated. This integration halves the power consumption and capacitor area typically allocated to the S/H circuit.
- *Simplification of Circuit Design*: Removing the S/H circuit simplifies the overall architecture of the Pipeline ADC. It reduces the number of components involved in the signal path and potentially improves the overall signal integrity by minimizing the noise contributions from the S/H stage.

Challenges of Embedding Sampling into the First Stage

- *Increased Complexity*: The integration of the sampling function into the first pipeline stage requires careful design considerations. The first stage must now perform dual functions: initially sampling the input signal and subsequently processing it through the MDAC. This added complexity necessitates precise timing and synchronization to ensure accurate signal acquisition and processing.
- *Impact on Timing and Accuracy*: Timing becomes critical when embedding sampling into the first stage. The stage must complete sampling before the MDAC switches to its amplification mode. Failure to synchronize these operations properly can lead to timing errors and degrade the overall accuracy of the ADC.

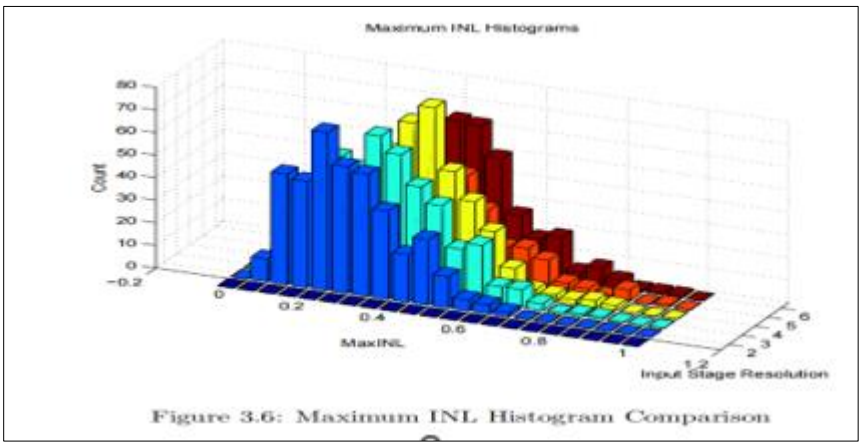


Figure 29 Maximum INL Histogram Comparison

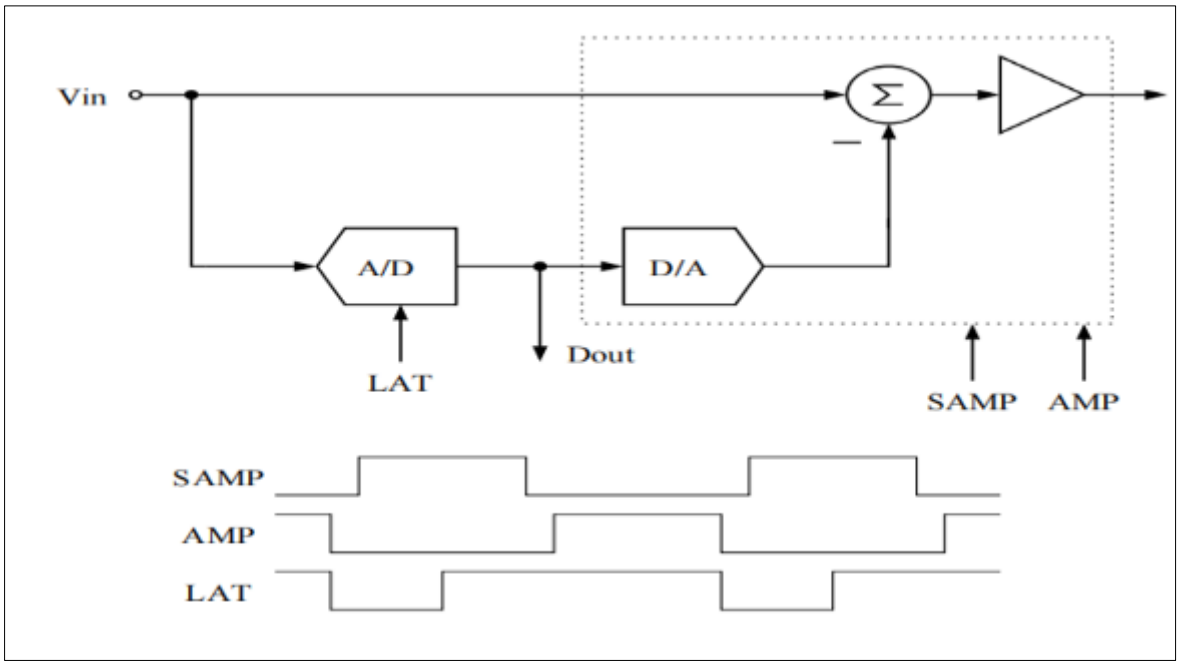


Figure 30 Pre-Resolution within Pipeline sub-ADC

The continuous-time input signal flows directly into the initial stage of the Pipeline ADC, with both the comparator and MDAC simultaneously capturing the input signal to achieve precise resolution of the residue voltage (as portrayed in Fig. 31). To address this requirement, a global input switching network was implemented, as discussed in [Mehr, I., & Singer, L. (2000).], in an effort to synchronize the sampling paths (as illustrated in Fig. 32). However, this solution still faces challenges associated with autonomous sampling switches and the impedance matching difficulty for both sampling paths. Specifically, it requires aligning the transconductance (Gm) of the comparator preamplifier and the Gm of the MDAC operational amplifier to ensure consistent transient responses across the sampling capacitors in the comparator and MDAC. Furthermore, it mandates separate reference sampling capacitors in the MDAC, which detrimentally impacts both the feedback factor and introduces kT/C noise.

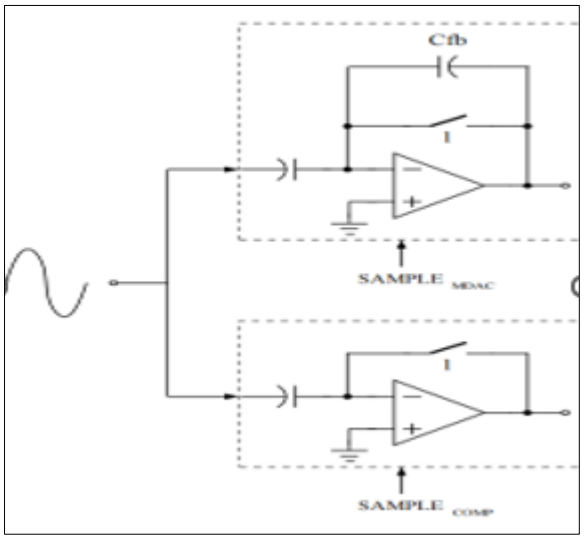


Figure 31 Two Sampling Paths

An enhancement to this architecture was introduced in [Chang, D. (2005).]. Instead of utilizing unity-gain feedback to sample the top plate of the sampling capacitor, both paths are sampled through MOSFET switches to a reference voltage (as depicted in Fig. 33). This modification simplifies the task of aligning the sampling paths. Additionally, due to the use of independent input switches, the need for an extra reference sampling branch in the MDAC is eliminated, and the kickback suppression provided by preamplifiers in the comparator is no longer required.

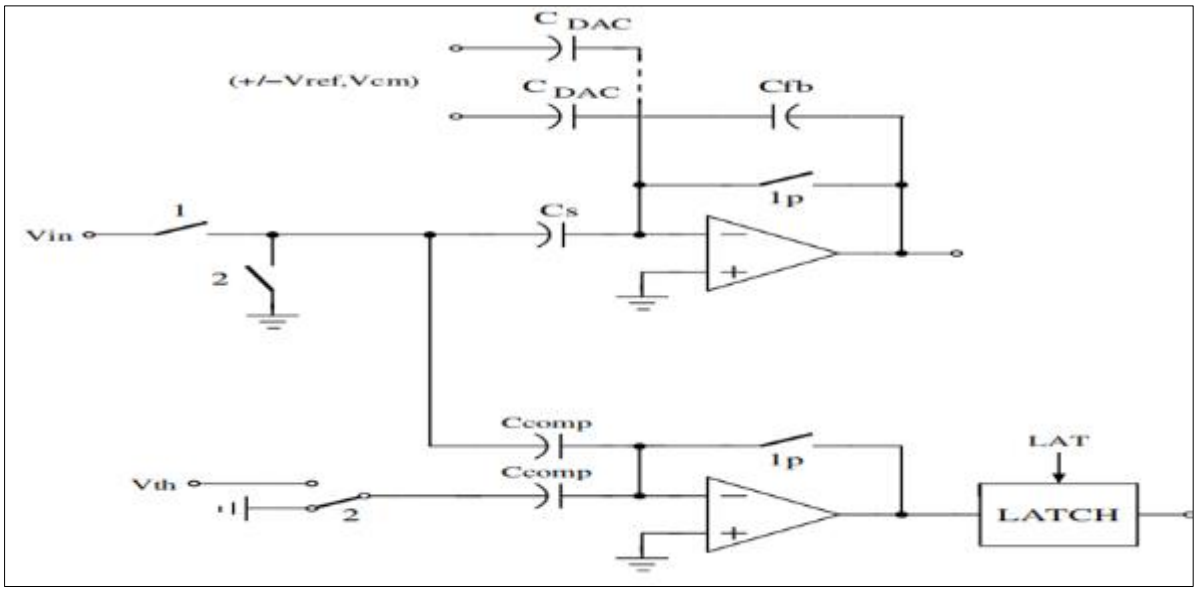


Figure 32 Shared Sampling Architecture without S/H

It is pertinent to state that a charge summing architecture was employed to carry out the comparison operation within the comparator. This was an imperative design choice due to the impracticality of pre-sampling the reference signal on the input capacitor. Additionally, careful consideration must be given to latch regeneration time, particularly in the initial stage of the pipeline ADC. In a traditional Pipeline ADC equipped with a Sample-and-Hold stage, the latches within the first-stage comparators regenerate during the interval between the comparator's sampling moment and the MDAC's (Multiplying Digital-to-Analog Converter) sampling moment. This ensures that the reference signals destined for the MDAC's DAC (Digital-to-Analog Converter) capacitors, used during the amplification phase, are precisely determined and stabilized before the MDAC switches modes.

However, with the elimination of the Sample-and-Hold stage, the scheduling of latch regeneration time becomes necessary in the interim between the dual-channel (MDAC and Comparator) sampling phase and the subsequent amplification phase. This introduces an additional phase into the process and, given an assumption of a uniform distribution of the conversion period, unfortunately leads to an undesirable reduction in processing speed.

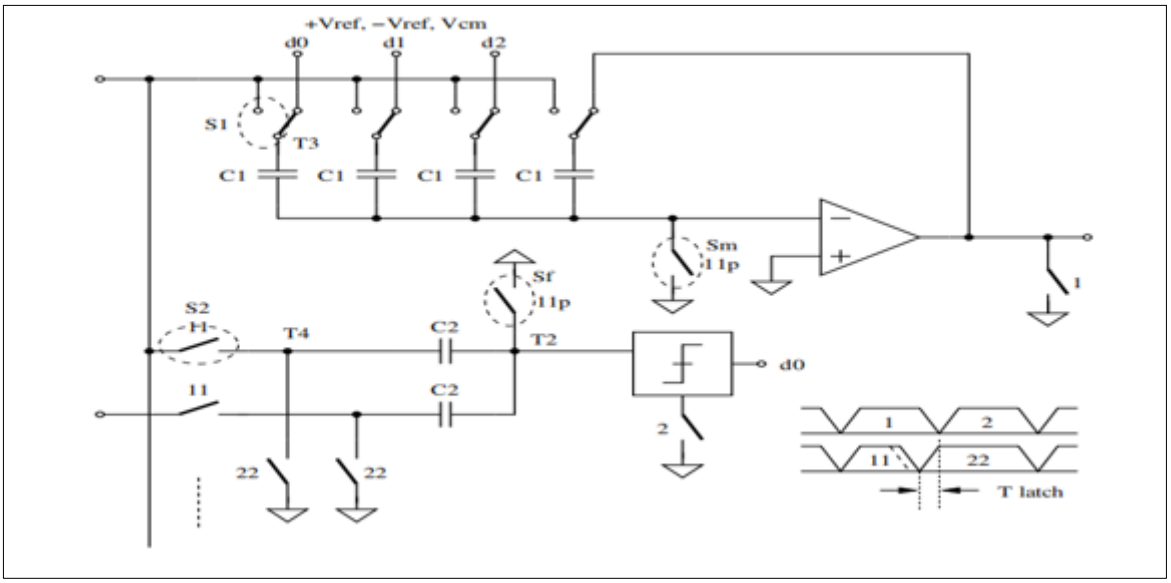


Figure 33 Separate Sampling Architecture without S/H

A straightforward approach to address this challenge is to incorporate the latch regeneration duration into the sampling interval. Given that the sampling switches need to be finely tuned for optimal linearity, they inherently possess relatively

short time constants, ensuring rapid settling times. Prioritizing the reduction of amplification time over the reduction of sampling time proves to be more crucial, particularly when considering the potential increase in power consumption required by the amplifier.

In this work, i employed an adapted version of the Sample-and-Hold (S/H) removal concept presented in reference [Chang, D. (2005)]. This adaptation involves retaining preamplifiers within the comparators and utilizing output offset storage (OOS) techniques [Razavi, B., & Wooley, B. (1992).] to nullify both latch and preamplifier offsets. This becomes especially significant since the offset of the MDAC opamp, coupled with any sampling discrepancies, can encroach upon the digital correction range. Figure 3.15 illustrates the implementation, complete with timing considerations that encompass the latch regeneration period.

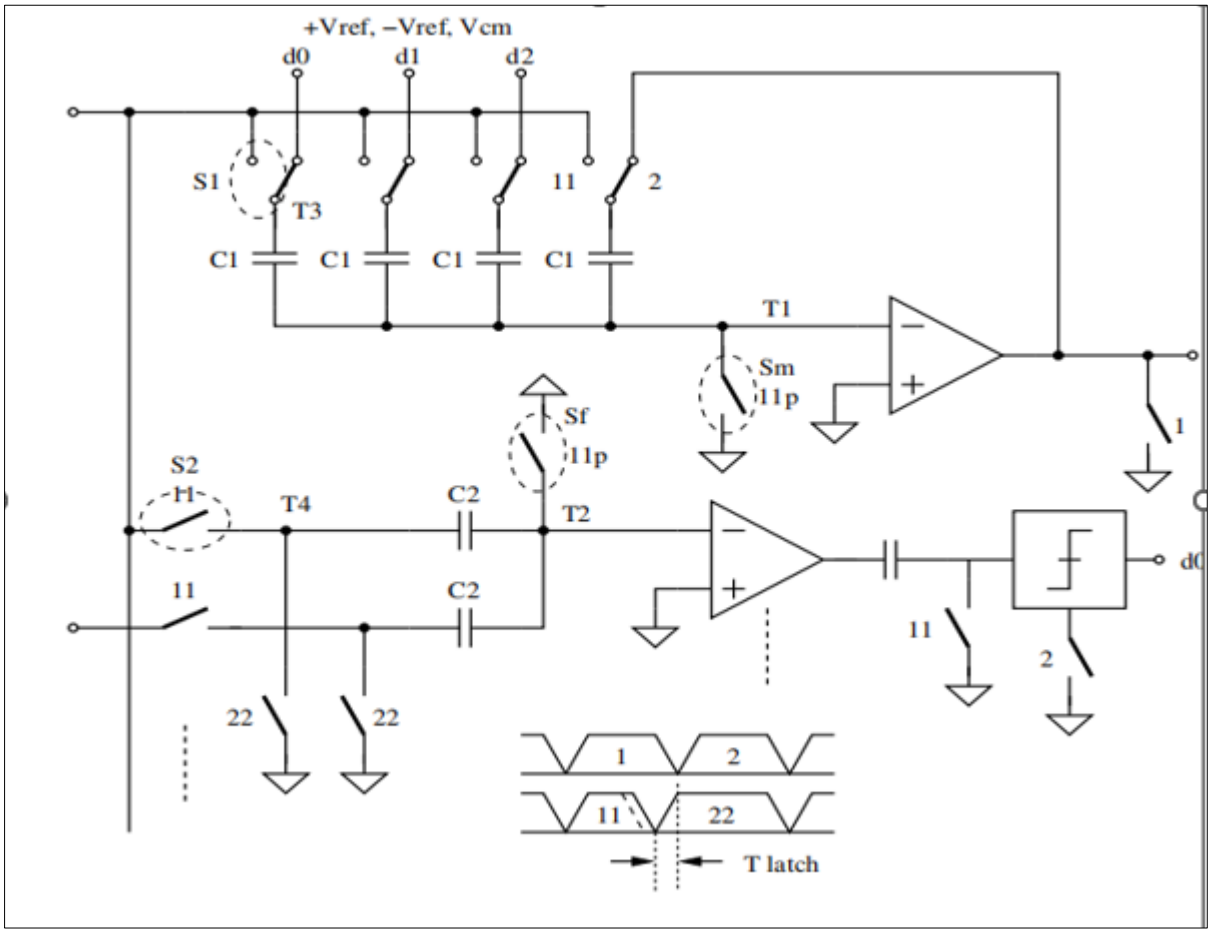


Figure 34 First Pipeline Stage with no S/H

3.7.2 Mismatch Tolerance

The crucial demand for the initial sampling stage lies in the precision of both the comparator input sample and the MDAC input sample. They must reside within the digital correction range of the pipeline stage, typically within 1/2 LSB. Accomplishing this precision involves two key aspects: harmonizing the continuous-time responses of the RC circuits and ensuring harmonization during the sampling operation (switching). Addressing the first aspect involves striving for the utmost precision in harmonizing the RC networks. However, achieving precise harmonization is an arduous challenge due to physical variations in device placement on the silicon die. Moreover, the comparator incorporates a capacitive branch that isn't part of the MDAC, utilized for charge summation during comparison. These factors impose limitations on achieving flawless harmonization for the continuous-time signals. One potential approach is to employ unit-sized capacitors in both the comparator and MDAC, which can enhance capacitor harmonization between their respective sampling capacitors. Nevertheless, this solution isn't ideal due to the substantial disparity in desired capacitance sizes between the two paths.

An equation quantifying the permissible RC mismatch (to a first-order approximation) is presented in [12]:

V-error = V · 2πfinτ ^{2}, (3.18)

Here, V-error represents the error voltage, V signifies the signal amplitude of a sine wave, τ is the RC time-constant of the assumed first-order system, and ^{2} denotes the time-constant mismatch. This error voltage can be conceptualized as an offset and is amenable to correction if it remains within the 1/2 LSB correction range of the stage. It's notable that the error voltage escalates with input frequency. Assuming a settling time of 10 · τ for 15-bit settling, finτ < 0.1 is valid for all input frequencies below 40MHz (assuming a 25nS settling time). Under this assumption, equation 3.18 can be reimagined with respect to ^{2} as:

$$\epsilon < \frac{0.5 \cdot LSB}{2\pi \cdot 0.1}. \qquad (3.19)$$

For a 3.5-bit-per-stage system, the requisite ^{2} matching for correction through digital redundancy is approximately 5%. Accomplishing this level of matching between different types of components is challenging, necessitating the use of similar components in both sampling paths. For instance, matching the transconductance of an amplifier to the 'on'-resistance of a MOSFET at a 5% level is a non-trivial endeavour. To further validate the allowable offset in a real multi-order system, SPICE simulations were executed. These simulations entailed gauging the deviation between the continuous tracking of the sampling capacitors and an input signal.

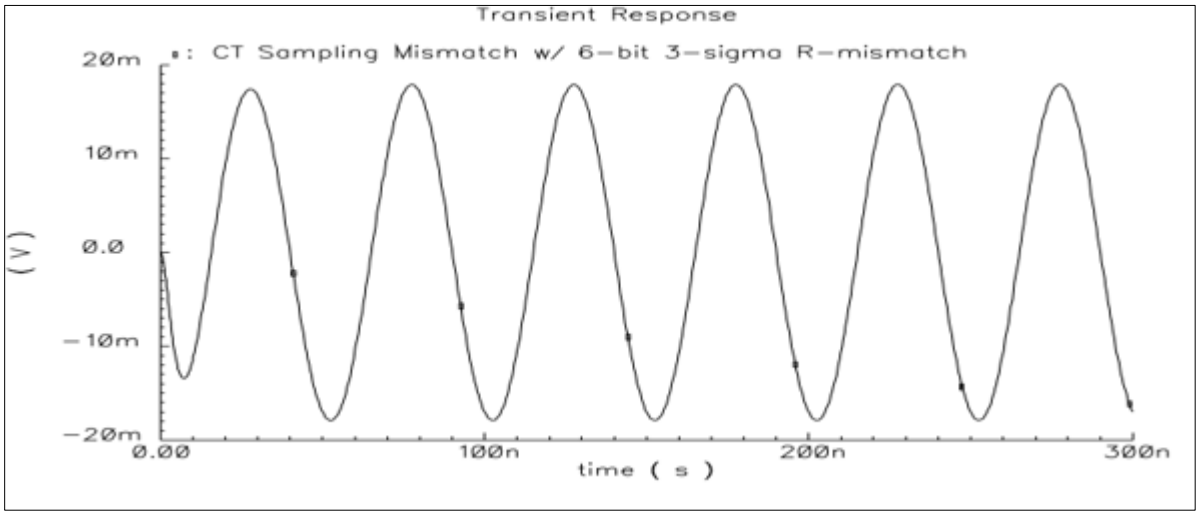


Figure 35 Continuous-Time Sampling Mismatch at 20MHz

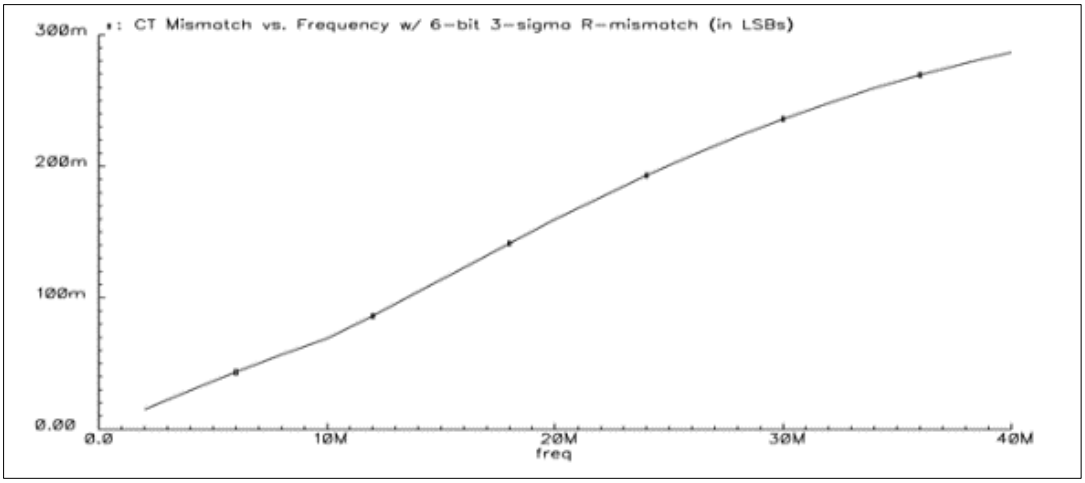


Figure 37 Continuous-Time Sampling Mismatch vs Frequency

In the context of achieving rail-to-rail (RTR) input capability in the Pipeline ADC, several key design considerations and modifications are necessary to ensure optimal performance and compatibility with the extended signal range:

3.8 Redesigning the mdac for rail-to-rail input

1. **Comparator Operation**: With the removal of the sample-and-hold (S/H) circuit, the input signal range is no longer constrained by the previous limitations imposed by the S/H circuit. Instead, it now adheres to the power supply voltages. For a true rail-to-rail input operation, the comparators within the Pipeline ADC's first stage must be designed to accommodate the full range of the supply rails. This adjustment ensures that the comparators can accurately process signals that span from the lower to the upper rail voltages.

2. **MDAC Reference Voltage**: In traditional Pipeline ADC designs, the MDAC reference voltage is crucial for accurate signal processing. With the adoption of a rail-to-rail input architecture, the MDAC's DAC segment must generate reference voltages that correspond to the supply rails. During the amplification phase of the MDAC, these references are injected to accurately scale and process the input signal. This requires careful adjustment of the MDAC's internal references to match the full range provided by the power supply voltages.

3.9 Implementing Dual-Reference Design

- Precision in Output Voltage Levels: To effectively utilize rail-to-rail input capabilities, the MDAC output stage needs to precisely drive signals towards the power supply rails, typically denoted as VREF2. Ensuring accurate voltage levels at these rails is critical for maintaining signal integrity and achieving high resolution throughout the ADC conversion process.
- Challenges and Solutions: One challenge in this redesign is ensuring that the amplifier responsible for setting the output levels can accurately reach and maintain the supply rail voltages without introducing significant errors. Precision in reaching these rails directly impacts the overall performance of the Pipeline ADC in terms of signal accuracy and noise performance.

By adopting a rail-to-rail input approach and redesigning the MDAC accordingly, the Pipeline ADC can benefit from an extended input signal range that aligns with the power supply voltages. This enhancement not only conserves power by eliminating the S/H circuit but also enhances the signal-to-noise ratio (SNR) and contributes to achieving the targeted Effective Number of Bits (ENOB). However, careful attention to detail in circuit design, particularly in setting reference voltages and ensuring precise comparator operation, is essential to maximize the benefits of rail-to-rail input while maintaining overall ADC performance. These efforts are crucial for meeting the stringent requirements of high-resolution ADC applications in modern electronics.

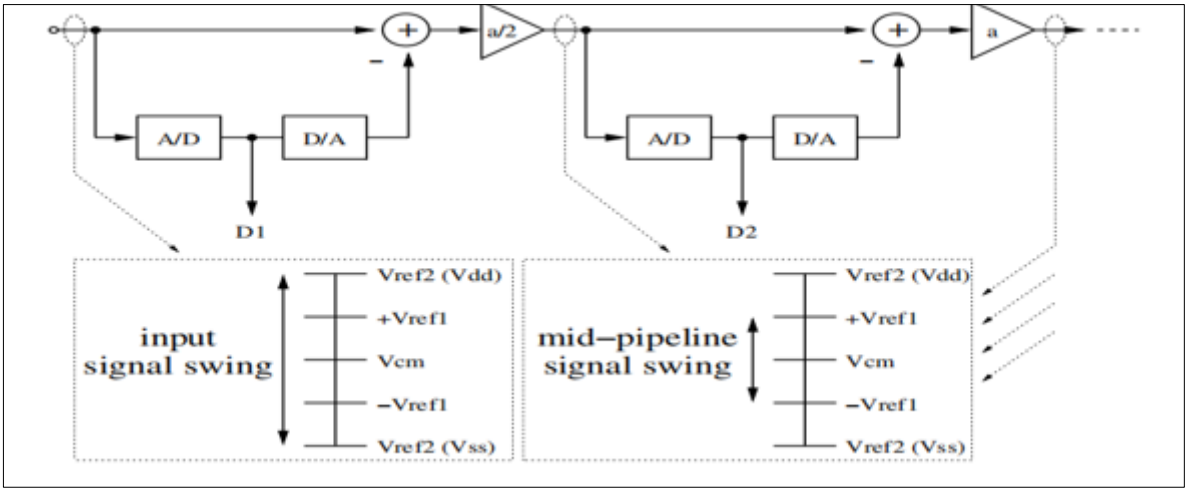


Figure 38 Two-Reference Pipeline Architecture

A straightforward remedy to this issue involves reducing the MDAC gain from the standard 2^(n-1) to 2^(n-2), where 'n' denotes the number of bits resolved in the current stage. Consequently, the remaining pipeline stages would operate with a reference voltage equivalent to half the power supply rails, or VREF1. The final architecture is depicted in Figure 38.

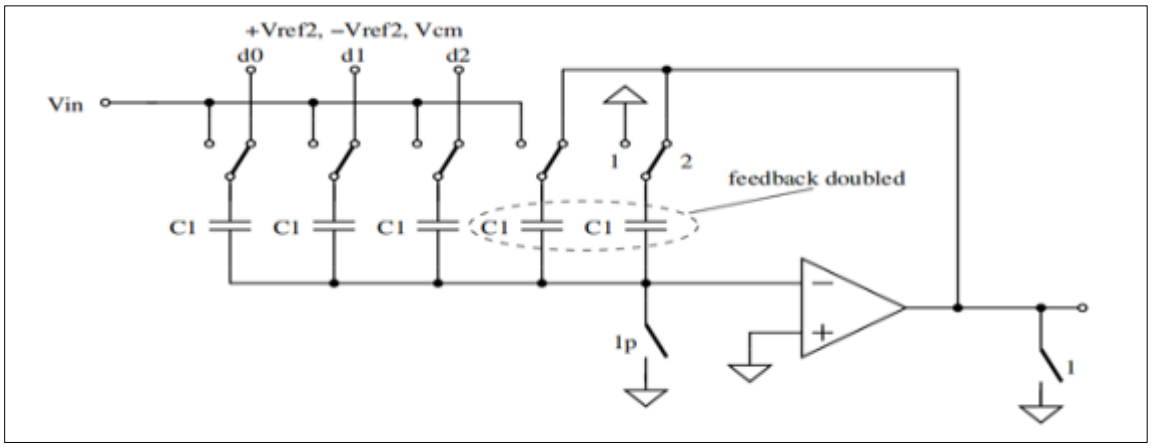


Figure 39 Final Stage MDAC Redesign for Two-Reference Pipeline

Implementing a single-reference design for a rail-to-rail input Pipeline ADC involves setting up all stages to operate with either VREF2 or VREF1 reference voltages. Here's how this approach unfolds:

- **First Stage (VREF2):** The initial stage of the Pipeline ADC remains largely unchanged from a standard MDAC structure, except for adjusting the feedback capacitance to accommodate the rail-to-rail input requirement. This stage utilizes VREF2 as its reference voltage, ensuring it covers the full range of the power supply rails.
- **Subsequent Stages (VREF1):** To maintain continuity and ensure proper operation across all stages, subsequent pipeline stages are modified. These stages are configured to operate with VREF1, which is set to half the voltage of VREF2. This adjustment necessitates recalibrating the capacitance values in these stages to reflect the lower reference voltage, thereby ensuring consistent performance across the entire ADC.

By adopting this single-reference strategy, the Pipeline ADC simplifies the implementation of rail-to-rail input capability while mitigating potential issues associated with reference mismatch. This approach ensures efficient signal processing while maintaining alignment with the desired power supply voltage levels throughout the conversion stages.

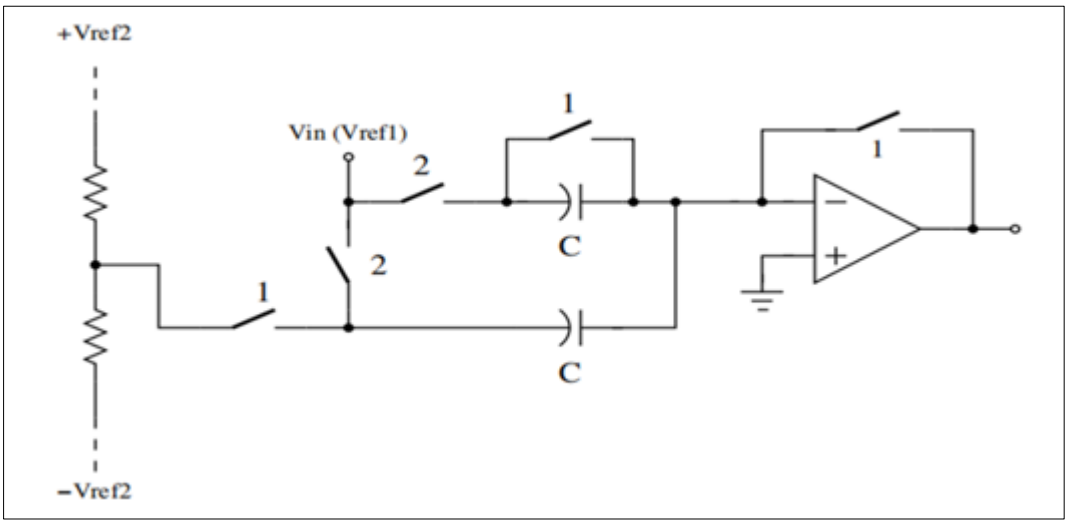


Figure 40 Double Design for Single-Reference Pipeline (V_{REF^2})

In the context of using VREF2 as the single reference for all stages of the Pipeline ADC, modifications are made to the MDAC to accommodate this setup. Specifically, the DAC section of the MDAC is adjusted by incorporating capacitors that are half the size of those typically used. This adjustment ensures that the cumulative charge subtracted from the signal remains consistent with the use of VREF1, despite using VREF2 as the sole reference voltage across all stages. Figure 3.21 illustrates a 2.5-bit implementation of this concept.

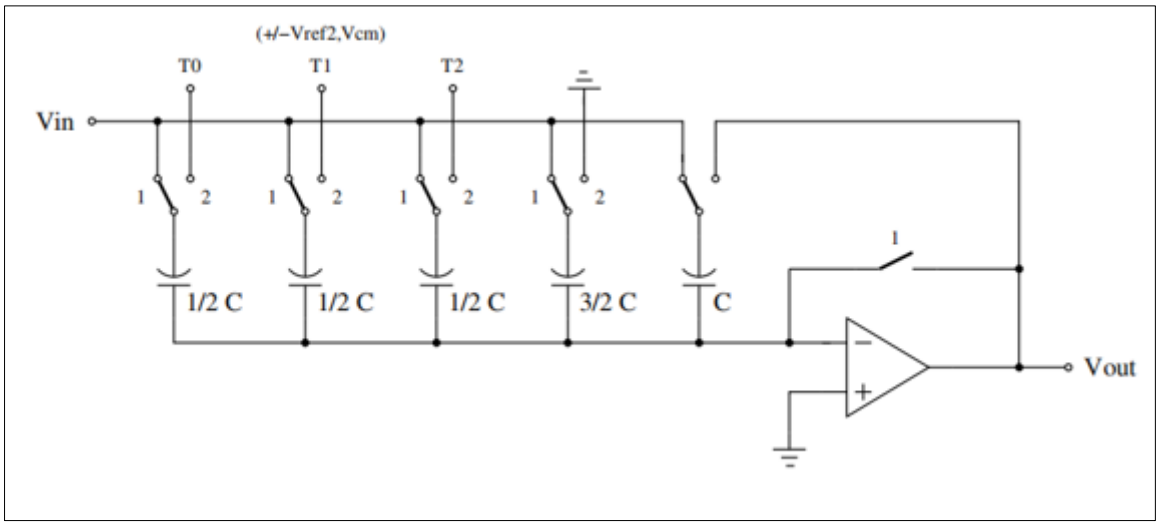


Figure 41 MDAC Design for Single-Referenc Pipeline (V_{REF}^2)

The second approach, employing a single-reference solution, involves utilizing VREF1 as the sole reference source for the entire converter. In this configuration, all stages subsequent to the initial one can employ standard Pipeline ADC stages. However, similar adjustments must be made to the first stage, akin to those applied to the remaining stages in the VREF2 single-reference solution.

For this setup, the comparator in the first stage necessitates weighted capacitors, similar to the other single-reference solution. As in the original S/H removal comparator block, the sampling in both the comparator and MDAC must be synchronized, requiring the implementation of a charge-summing structure. Notably, the sampling capacitor in the reference ladder must be twice the size of the signal sampling capacitor (as depicted in Fig. 42). This adjustment accommodates the disparity between the signal range and the available reference voltage. The additional capacitance serves to mitigate sampling inaccuracies between the MDAC sampling path and the comparator's sampling path.

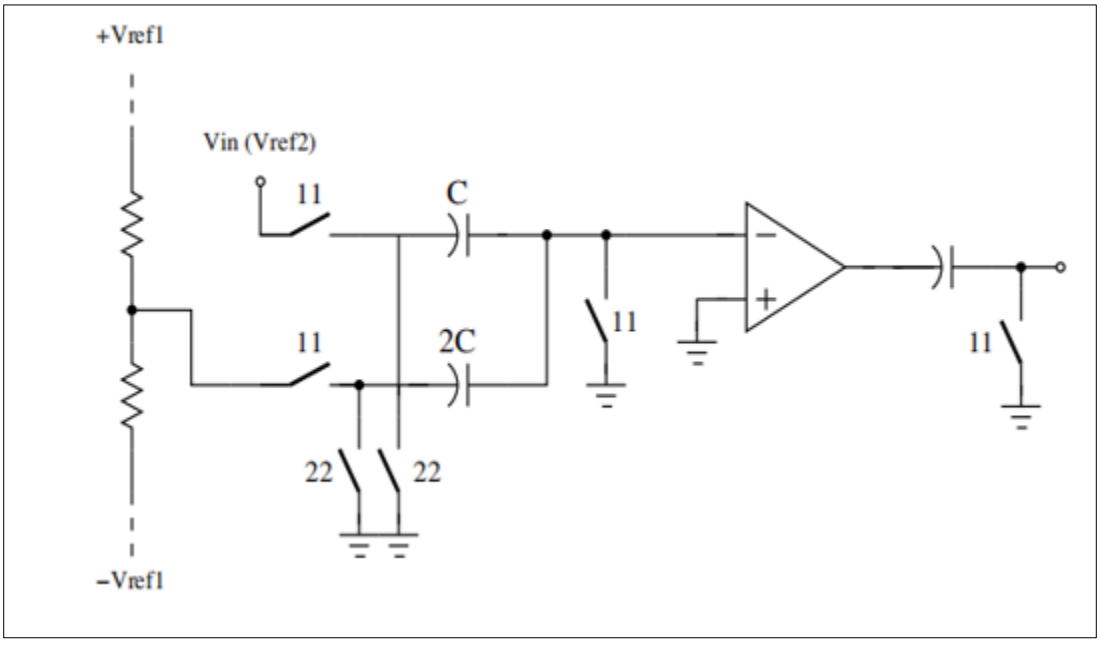


Figure 42 Comparator Design for Single-Reference Pipeline (V_{REF}^2)

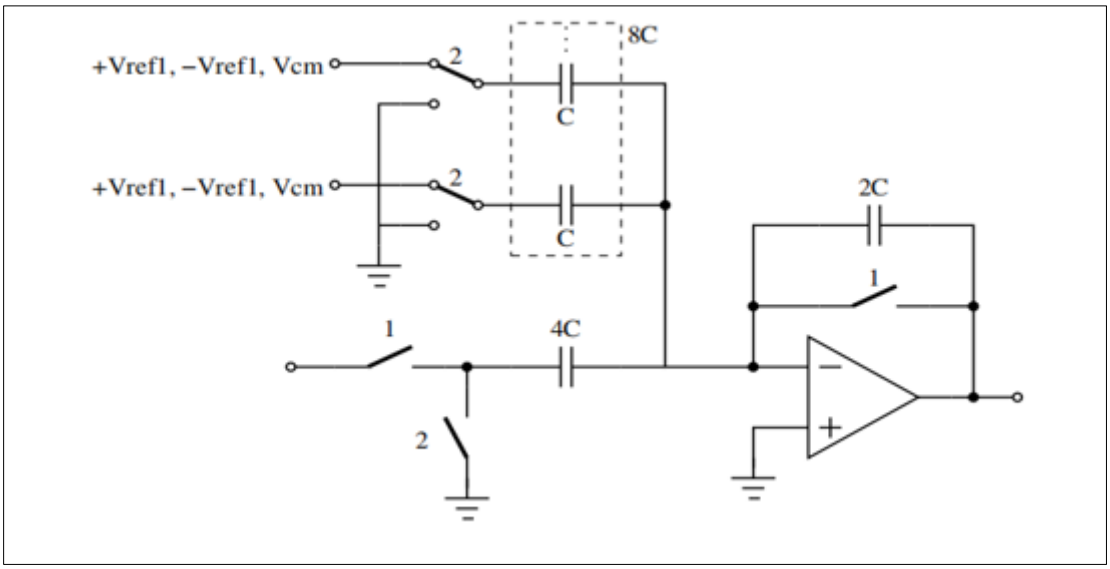


Figure 43 MDAC Design for Single-Reference Pipeline (V_{REF}^1)

The initial MDAC stage must exhibit a similar response as the subsequent stages when implementing the VREF2 single-reference solution. To achieve this, the capacitors within the MDAC must be doubled in size, effectively subtracting a signal with a VREF2 reference. The configuration of this MDAC is depicted in Figure 3.23. The introduction of additional capacitance at the input of the amplifier has the consequence of reducing the feedback factor, necessitating higher power consumption for equivalent speed. Furthermore, this supplementary capacitance introduces more charge noise into the signal compared to the simpler two-reference version of the MDAC. An enhancement to this design can be attained by reusing the sampling capacitors within the DAC, as illustrated in Figure 3.24. Nevertheless, challenges related to feedback factor degradation and increased charge noise persist.

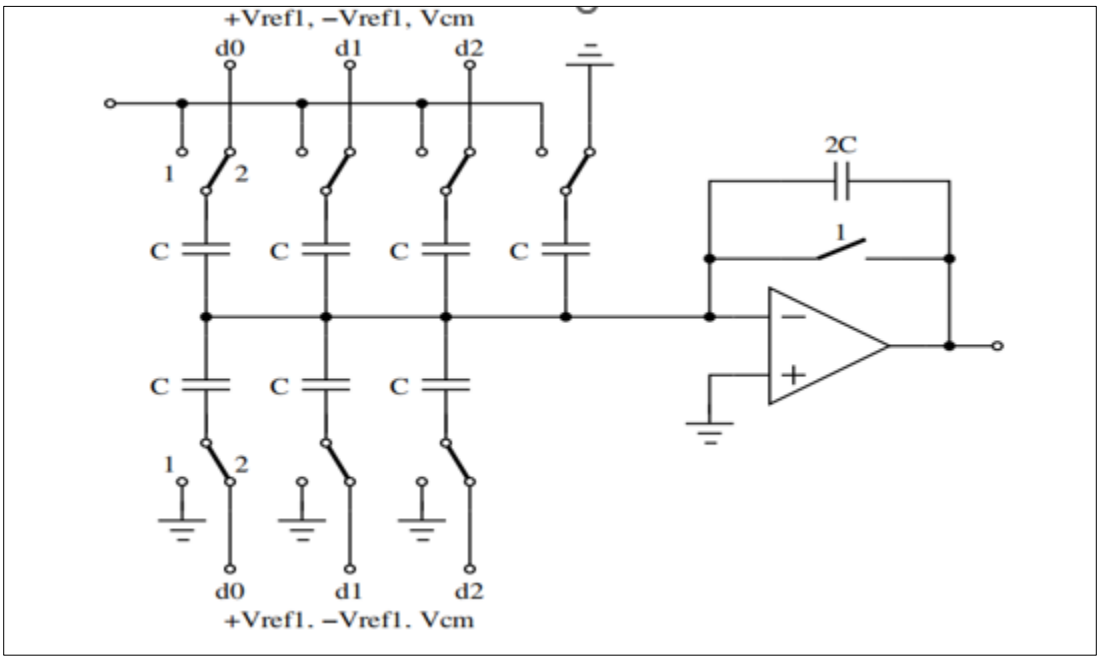


Figure 44 Better MDAC Design for Single-Reference Pipeline(V_{REF}^1)

Both single-reference solutions are viable options, but the VREF1 single-reference solution appears to be more advantageous. It avoids the drawbacks associated with the VREF2 single-reference solution, such as the kT/C and feedback factor penalties. Ultimately, the decision was made to adopt the two-reference solution due to its simplicity and the opportunity it provides to calibrate other errors concurrently with the reference mismatch calibration. In the context of reference calibration, it was previously noted that any deviation between the two reference levels within the

two-reference, rail-to-rail solution could result in non-linear errors. Moreover, this error arises from a gain-related issue occurring at the boundary between the initial stage and the subsequent stages.

As detailed previously, gain errors can be rectified through the implementation of a radix calibration scheme [. Karanicolas, A., Lee, H., & Bacrania, K. (1993).]. This correction process involves two essential steps: firstly, determining the actual radix, and secondly, utilizing this determined radix to compute the final resolved bit values. It's worth noting that this calibration method can be computationally intensive, which may impose limitations on its practical application. Figure 45 illustrates a circuit method employed for gain error extraction. It can be demonstrated that any increment in the one-bit code from (a) to (b) yields an output voltage equivalent to (1/4) of VREF2, assuming an input range spanning between +/- VREF2. The equation that characterizes the relationship between VREF1 and VREF2 takes the form:

$$VREF2 = \beta \cdot 2VREF1, \quad (3.20)$$

In the context of β, equation 3.20 can be rewritten as follows:

$$\beta = (1/4) \cdot VREF2 \,/\, ((1/2) \cdot VREF1) = DHR \cdot VREF1 \,/\, ((1/2) \cdot VREF1) = 2 \cdot DHR, \quad (3.21)$$

Here, DHR represents the digital half-reference, as determined by the subsequent stages within the pipeline ADC. Its computation is relative to the reference employed in those stages, which is VREF1. The final value of beta (β) is derived by averaging multiple measurements. Subsequently, a floating-point multiplier is utilized to calculate a correction code for each first-stage code. These correction codes are then stored in RAM memory, utilizing the first-stage code as the address and the correction code as the stored value. This process serves as an initial task for a system equipped with an on-board processor capable of allocating resources for an initial calibration cycle.

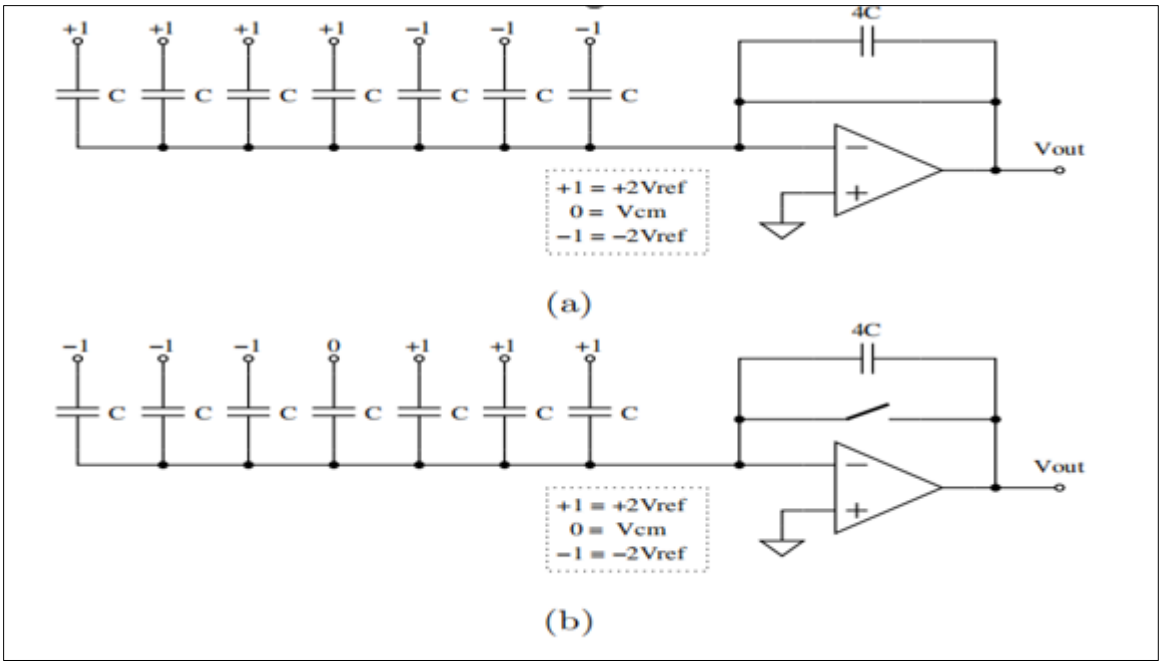


Figure 45 Reference Mismatch Extraction Circuit (Two Phases**)**

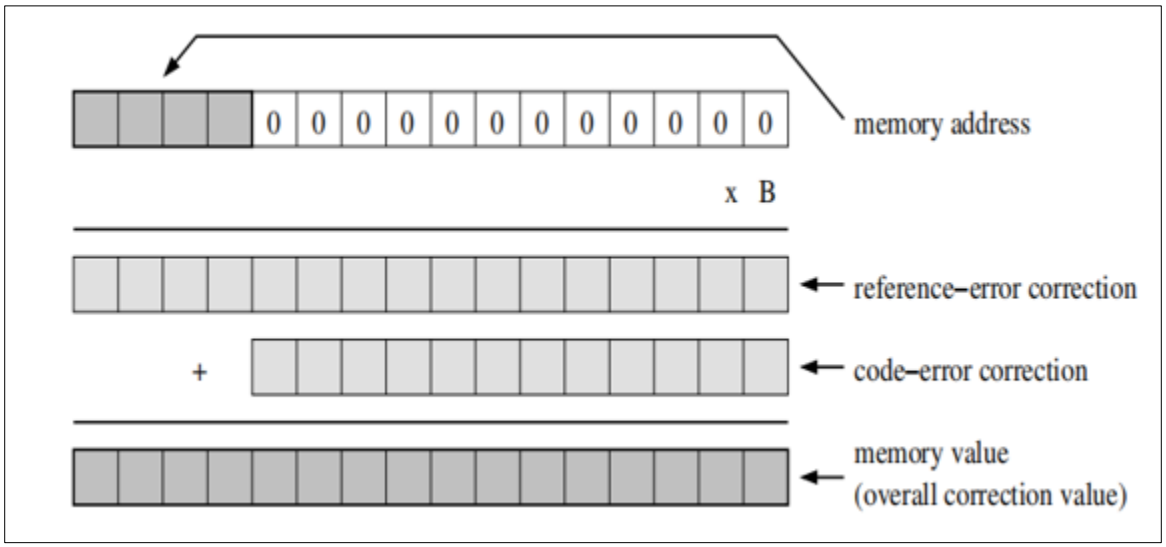


Figure 46 Generation of Correction Code

The measurement circuit for reference mismatch closely resembles the method used for error extraction in code-error calibration [21]. It is plausible to utilize this circuit for code-error calibration when the need arises, particularly to counterbalance capacitor mismatch in the initial stage of the pipeline. Figure 46 illustrates the process of determining and storing the error correction code. In Figure 47, the simulated enhancement in the linearity performance of a Pipeline ADC is depicted, taking into account approximately 0.7% reference mismatch and β, which is measured using 13 bits of quantization from the ADC's backend.

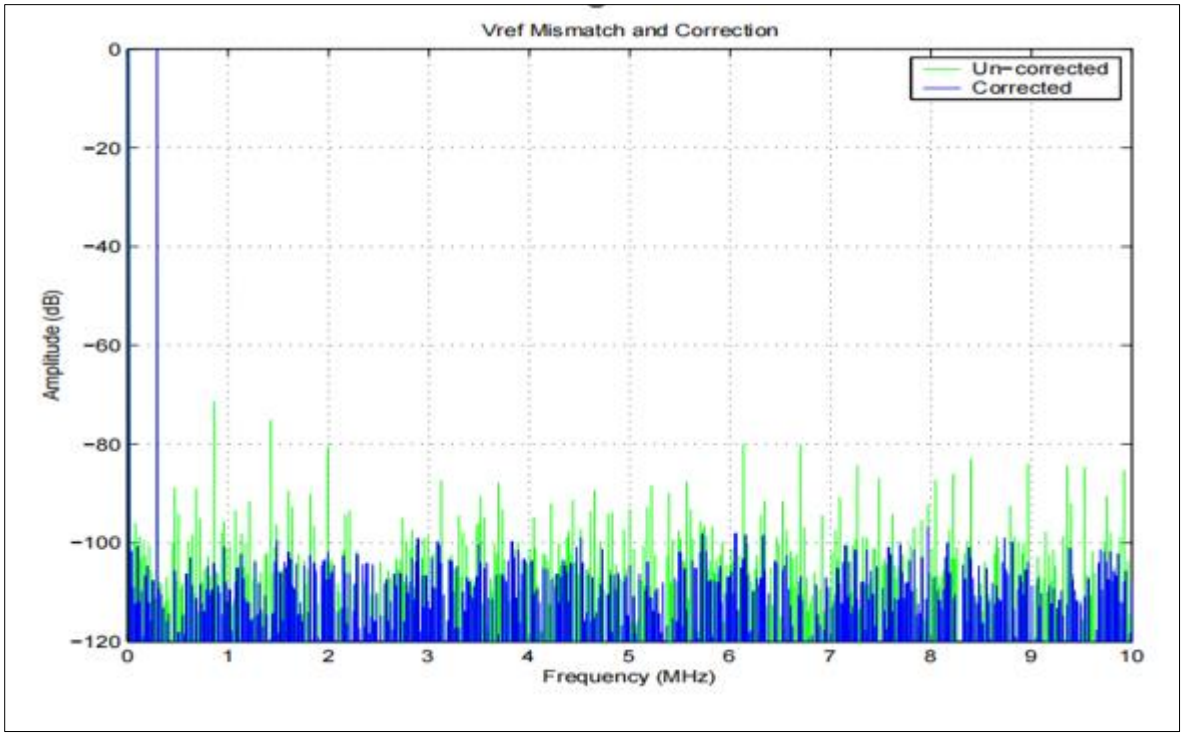


Figure 47 MATLAB-Simulated Reference Mismatch Calibration FFTs

3.9.1 Noise budget estimation

Before initiating the actual circuit design, determining the unit capacitors for each stage in the pipeline becomes essential. This calculation hinges on achieving the desired Signal-to-Noise Ratio (SNR), factoring in kT/C thermal noise

contributions, and the taper factor. To begin, it's imperative to compute the allowable pipeline input-referred noise attributed to kT/C thermal noise.

Step 1: Calculating SNR Ratio

The initial step entails the computation of the Signal-to-Noise Ratio (SNR) ratio. This ratio is assessed at the output of the first stage due to its practicality in aligning with the optimization scheme described in Section 3.4. The SNR at this juncture should be three bits lower than the overall pipeline SNR since the first stage already resolves these bits without redundancy.

Mathematically, this is expressed as:

SNR=6.02·3=18.06 dB≈18 Db

For convenience, this value is rounded up to 18 dB and converted into a "Voltage" measure:

SNR=18 dB=6.309×102

Step 2: ***Calculating Input Signal Power***

The next crucial step involves the calculation of input signal power $\sigma u2$ This can be determined using the formula:

$\sigma u2=(2FS)22=0.922=0.8464$

Here, FS represents the maximum differential signal swing at this specific circuit point.

Step 3: ***Determining Allowable Output Noise***

Combining the SNR and input signal power computations, we can derive an equation for the allowable noise, referred to the output of the first stage:

$\sigma u2\cdot\sigma n2=SNR\Rightarrow\sigma n2=SNR\sigma u2=6.309\times1020.8464=1.34\times10-3(\mu V)2=1.16\times10-6$ V2

Step 4: ***Consideration of Noise Sources***

The noise signal (\(σ^2_n\)) encompasses various noise sources, including jitter noise, active circuit noise, and kT/C sampled thermal noise. In this design, it's expected that kT/C thermal noise will be the predominant contributor to error. Therefore, a conservative 50% budget is allocated for kT/C thermal noise within the allowable noise.

This results in a total allowable noise voltage of 1.16×10−6 (μV)2 1.16×10−6(μV)2 or 1.08×10−3 μVrms 1.08×10−3μVrms at the output of the first stage

This calculated value is subsequently applied to the final architecture, incorporating a taper factor of 1.1 from one stage to the immediately following stage, spanning across all stages in the design.

STAGE	Ceq	NOISEout	Cunit	Cfb	Cs
1	484 fF	185 μV-rms	1.1 pF	2.2 pF	8.8 pF
2	49.2 fF	580 μV-rms	0.4 pF	0.4 pF	3.2 pF
3	13.5 fF	1151 μV-rms	0.1 pF	0.1 pF	0.8 pF
4	13.5 fF	1151 μV-rms	0.1 pF	0.1 pF	0.8 pF

Figure 48 Noise and Capacitor Stage Distribution

Figure 48 presents the selection of sampling capacitors and the noise contribution at the output node of each stage. In this context, 'Ceq' is defined as β·CF. The calculation of output noise can be expressed as v2n(kT /C) = 4kT·Ceq (Equation 3.31). The factor of '4' in the kT multiplier originates from the cumulative noise summation across two phases and an additional 2x summation from the combination of positive and negative differential paths. The cumulative output-referred kT/C noise for the first stage is measured at 199.5μV-rms. This stage structure serves as the foundation for the subsequent circuit design discussed in the following sections. The resolution and capacitor sizes at each stage act as constraints, streamlining the circuit design process and shaping various design specifications. These specifications encompass open-loop gain, feedback factor, capacitive load, and digital requirements for the calibration measurement control circuits and digital correction circuitry.

4. Results and discussion

In the preceding sections, a pivotal decision was made regarding the system architecture. This choice hinged on a comprehensive evaluation of power and area optimization factors. Ultimately, our selection led us to a two-reference Pipeline ADC with a per-stage resolution of 3.5 bits. Among these stages, the initial stage bears paramount importance in our design. The subsequent stages receive the input signal after it has undergone significant amplification, and some portion of the signal has already been resolved. Consequently, the demands placed upon the latter stages are considerably less stringent compared to the rigorous requirements of the first stage. Several specifications must be met by the first stage, including:

- Low Noise: We aim for a low noise level, approximately 180μVrms at the output, considering all noise sources.
- Linearity: Achieving linearity is crucial, with a target of more than 15 bits of linearity during the sampling phase and more than 12 bits of linearity at the output.
- Settling Time: For both sampling and output, we strive for more than 14 bits of settling during sampling and more than 11 bits of settling at the output.

The sampled thermal noise component of the overall noise has already been factored in through the optimization process detailed in the preceding sections. The linearity depends on the precision of the components and the gain of the open-loop amplifier. Settling time is influenced by the pole locations within the circuits and can be determined through the use of RC-networks as well as the bandwidth and stability of the operational amplifiers.

4.1 First stage sampling

In previous section, the matching of both sampling paths was extensively discussed; however, details on the linearity requirements for the input sampling were not covered. Furthermore, the implications of rail-to-rail input on the input sampling circuits were not explored.

4.1.1 Switch Linearity

The linearity requirements of the sampling circuit are intricately tied to the overall desired linearity of the system. Consequently, the linearity of the first stage sampling must surpass the overall desired linearity. For a 14-bit Effective Number of Bits (ENOB) Pipeline ADC, we strive for first stage linearity of at least 15 bits. This ensures that the noise

component dominates the ENOB measurement. Moreover, if the linearity requirement is met for the entire system, then both the Differential Non-Linearity (DNL) and Integral Non-Linearity (INL) should also exceed 15 bits.

4.1.2 Implications of rail-to-rail input on input sampling circuits

The use of a rail-to-rail input range can introduce additional challenges in maintaining linearity within the input sampling circuits. These circuits must handle a wide range of input voltages while maintaining the required precision and linearity. This necessitates the use of high-quality switches and careful design to minimize any non-linear effects. The choice of switch technology, such as CMOS switches with low on-resistance and minimal charge injection, plays a crucial role in achieving the desired linearity. Additionally, the design must account for variations in process, voltage, and temperature (PVT) to ensure consistent performance across all operating conditions.

The design of the first stage of a high-resolution Pipeline ADC must prioritize low noise, high linearity, and fast settling time. By ensuring that the first stage sampling meets or exceeds these specifications, we can achieve the desired overall system performance. The elimination of the sample-and-hold circuit and the adoption of a rail-to-rail input further optimize the design, reducing power consumption and enhancing signal range, while requiring careful consideration of the linearity and matching of the input sampling circuits.

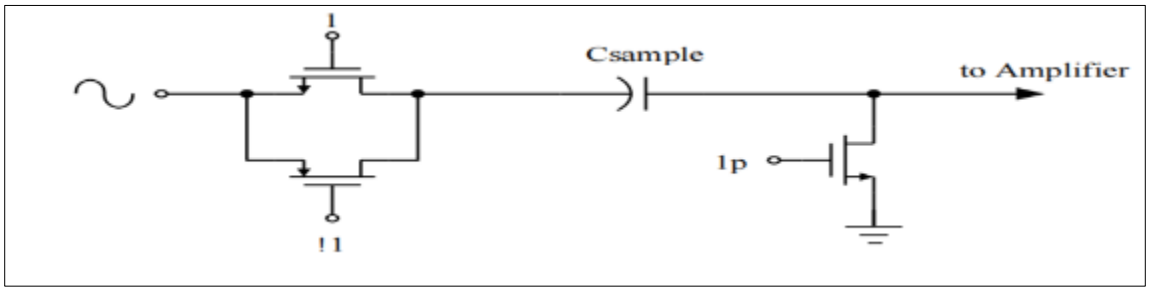


Figure 49 Input Sampling with CMOS Switch

Figure 49 illustrates a frequently used configuration for a switched-capacitor input sampling circuit. Within this setup, a 'floating' CMOS switch, often referred to as a transmission gate, allows the passage of signals across a wide range, spanning from the positive power supply to the negative power supply. It's important to note that this switch encounters alterations in its resistance that depend on the input signal, stemming from the transistor's 'on resistance,' which, in turn, relies on the gate-source voltage.

$$R_{ON} = \frac{1}{\mu_n C_{ox} \frac{W}{L}(V_{GS} - V_{TH})}, \qquad (4.1)$$

Here, μnCox represents a device constant, while W and L denote the physical dimensions of the MOSFET channel. Additionally, (VGS − VTH) signifies the MOSFET overdrive voltage. This signal-dependent resistance introduces non-linearity in the sampled voltage across the input-sampling capacitor. The impact of this non-linearity can be significantly mitigated by ensuring a balanced contribution of resistance from both PMOS and NMOS transistors. This balance enhances linearity since signals close to the positive power supply encounter comparable resistance from the CMOS switch as signals near the negative power supply. Achieving this equilibrium often requires the PMOS device to be 2 to 3 times larger than the NMOS device. In the context of the available 0.18um CMOS process with CMOS switches, achievable linearities are typically on the order of 10 bits (refer to Fig. 50)[MATLAB Code CMOS].

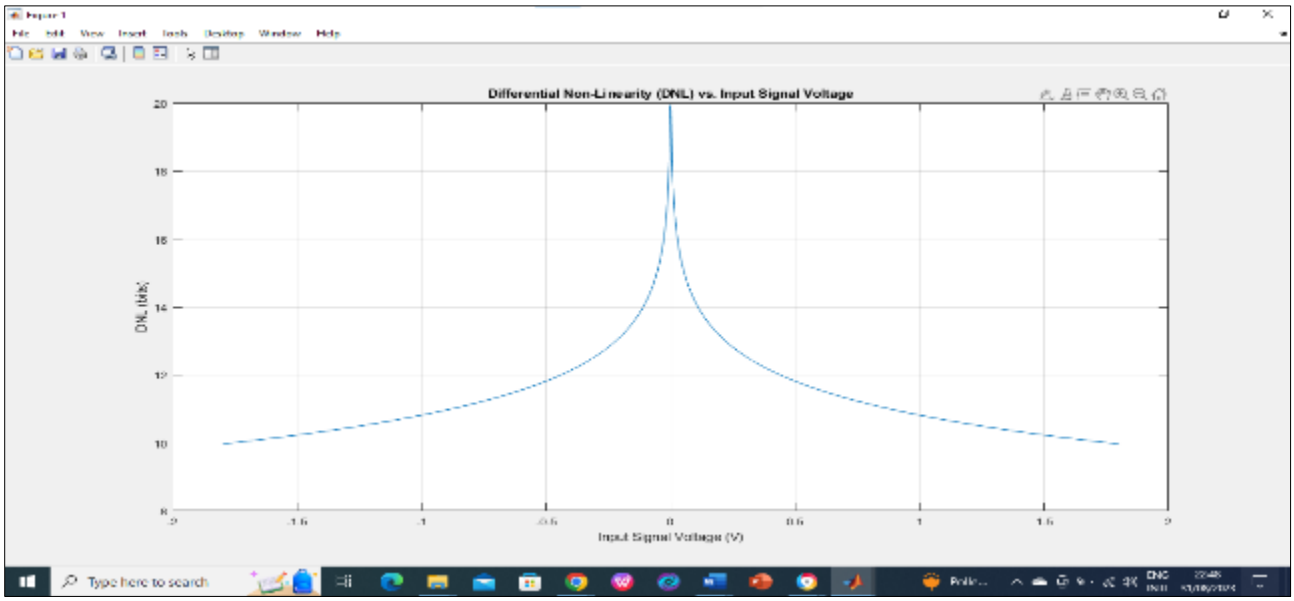


Figure 50 CMOS Switch Sampling Linearity

4.2 Bootstrapping

Achieving enhanced sampling linearity can be realized by maintaining a consistent gate-source voltage across all input signals. One straightforward approach involves inserting a voltage source between the input (source) and gate of an NMOS switch. It's worth noting that perfect voltage sources are typically unavailable within an integrated circuit (IC) design context. Therefore, capacitors are frequently employed as substitutes for the voltage source. Figure 4.3 illustrates a sampling system that incorporates a simplified form of bootstrapping.

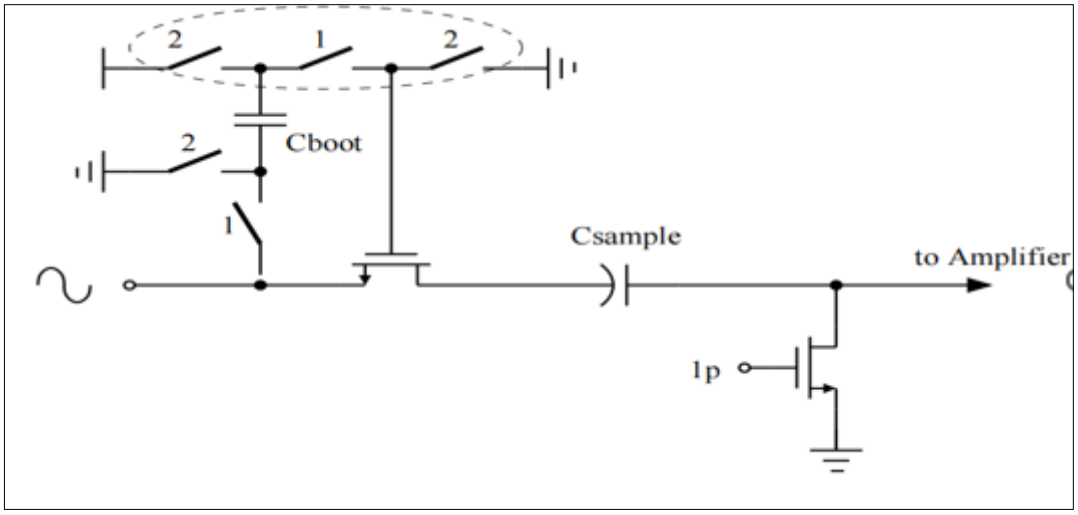


Figure 51 Input Sampling with Bootstrapped NMOS Switch

Addressing the challenge of bootstrapping clock circuits entails preventing all transistors from being exposed to voltages exceeding the power supply limit. For instance, the NMOS switch's gate switches (indicated in the diagram) confront source-bulk and drain-source voltages that surpass the power supply level. To overcome this issue, a widely adopted circuit, as introduced by Dessouky [Dessouky, M., & Kaiser, A. (2001).], was employed in this study. In the available 0.18µm CMOS process, a bootstrapped NMOS switch can achieve linearity levels of approximately 15 bits (as depicted in Fig. 51). The primary hurdle in the design of the first-stage MDAC opamp within this system was its multifaceted nature. It must not only adhere to stringent requirements for first-stage input and output linearity, settling, and noise, but also contend with a rail-to-rail input and sample a continuous-time signal.

The opamp must satisfy the following specifications:

- Open Loop Gain > 15 bits (equivalent to 90dB)
- Settling Accuracy > 12 bits
- Settling Time ≈ 25nS
- Linear Output Range = 900mV (single-ended)

Given the extensive output range (900mV while operating on a 1.8V supply) and the high-speed criteria of the system, the folded-cascode opamp stands out as a natural choice for the operational amplifier in the MDAC. The folding technique enables the attainment of maximum output range while preserving the high-speed simplicity of a single-stage amplifier. Figure 52 illustrates an example of how this amplifier will be utilized within the MDAC.

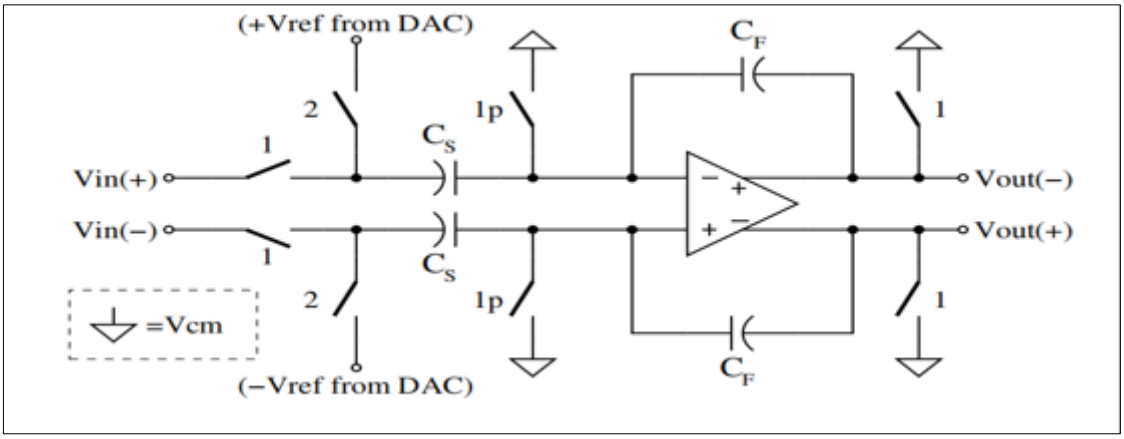


Figure 52 General MDAC Circuit

4.2.1 Gain Boosting

Attaining gains surpassing 80dB in a 2x "gmro" stage (single cascode) within state-of-the-art, short-channel transistor processes often proves to be unfeasible. Utilizing multiple cascodes is not a viable option to achieve the desired output range, and even two-stage cascode or folded cascode amplifiers would still suffer from the speed penalties associated with compensating a two-pole system.

In such circumstances, a valuable technique emerges—gain boosting, also known as active cascode [16]. The underlying concept of this method is clear-cut: an auxiliary amplifier is harnessed to propel the gate of the second transistor in a cascode configuration (as portrayed in Fig. 4.6). The amplifier guarantees the stability of the drain node of the first transistor, with accuracy inversely linked to the open-loop gain of the amplifier. This effect complements the resistance augmentation offered by the cascode transistor. Furthermore, gain boosting can be actualized in the guise of a fully-differential amplifier. In this context, the output common mode control of the amplifier establishes the bias voltage of the cascode transistor. The differential gain boosting approach was selected for this endeavor due to its ease of implementation and inherent symmetry properties.

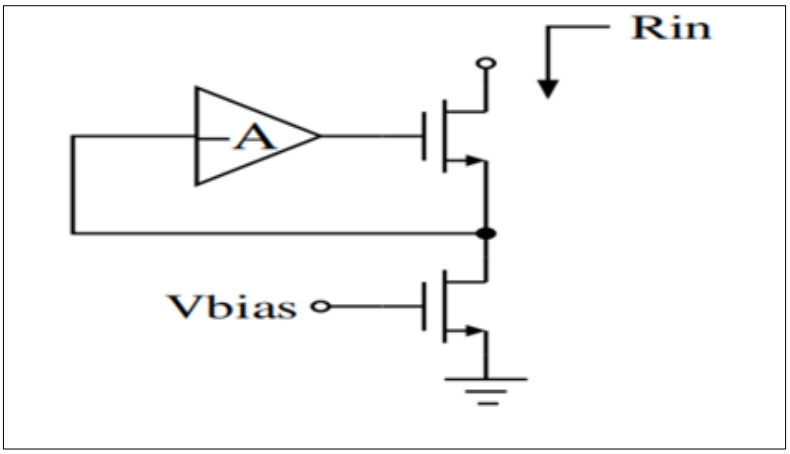


Figure 53 Active Cascode Example

An issue that can arise with the active cascode technique involves introducing a pole-zero doublet into the overall amplifier transfer function. This concern has been thoroughly addressed in prior research [43]. The findings of this study reveal that the pole-zero doublet can be effectively managed by maintaining a specific relationship among the amplifier poles. This relationship is defined as follows:

$$\beta\omega_{ugbw\text{-}main} < \omega_{ugbw\text{-}boost} < \omega_{p2}, \quad (4.2)$$

Here, β represents the feedback factor of the opamp system, $\omega_{ugbw\text{-}main}$ corresponds to the unity-gain bandwidth of the main amplifier, $\omega_{ugbw\text{-}boost}$ pertains to the unity-gain bandwidth of the boost amplifier, and ω_{p2} signifies the second pole of the main amplifier. The amplifier in question has been meticulously designed to satisfy these stringent criteria. Verification of this design was achieved through the observation of a transient step in the amplification phase configuration. The step response demonstrated stability, settling to an accuracy of approximately 14 bits within the desired 25nS half-clock phase duration.

4.2.2 Common- Mode Feedback

In addressing common-mode feedback within this amplifier, a dual-pronged approach was adopted. First, within the main amplifier, a common switched-capacitor feedback system was implemented (as depicted in Fig. 54 [16]). This circuit employs a capacitive divider to determine the common-mode level of the amplifier, and feedback is directly applied to the gate of a current source situated within the amplifier. During every non-amplifying phase, the capacitor charge is refreshed to prevent any drift in the common-mode value.

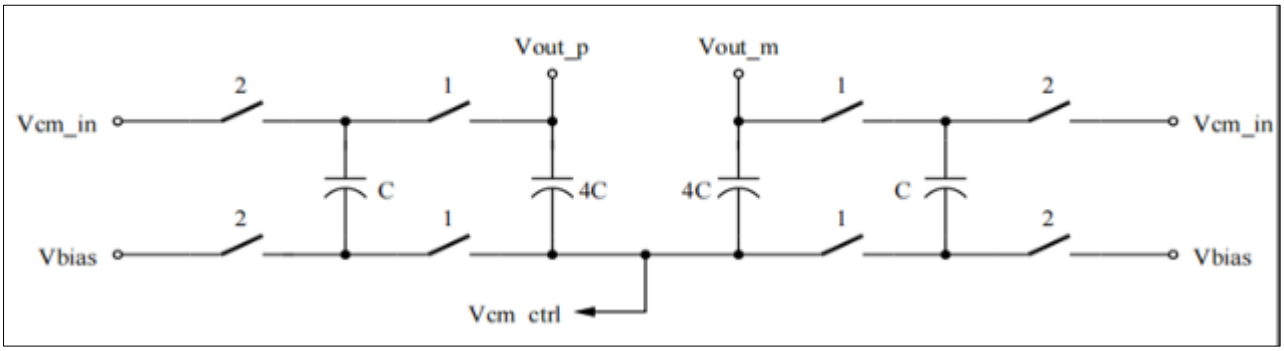


Figure 54 Switched-Capacitor Common-Mode Feedback Circuit

The amplifier employs a second method of common-mode feedback known as linear-region MOSFET common-mode feedback [Johns, D., & Martin, K. (1997)]. This CMFB technique proves particularly advantageous in systems with limited output swing, given the relatively narrow linear range of a MOSFET transistor. Consequently, the boost amplifiers serve as an ideal location for implementing this form of common-mode feedback. The outputs of the boost amplifiers only require slight adjustments to maintain their gain-boosting functionality.

4.2.3 Full Amplifier Architecture

Figure 4.8 illustrates the final architecture of the first-stage MDAC amplifier. It is worth noting that gain-boosting is applied to both sets of cascode transistors. Additionally, common-mode feedback control is implemented on the NMOS current sources since alterations in the current flowing through these components solely impact the output branch.

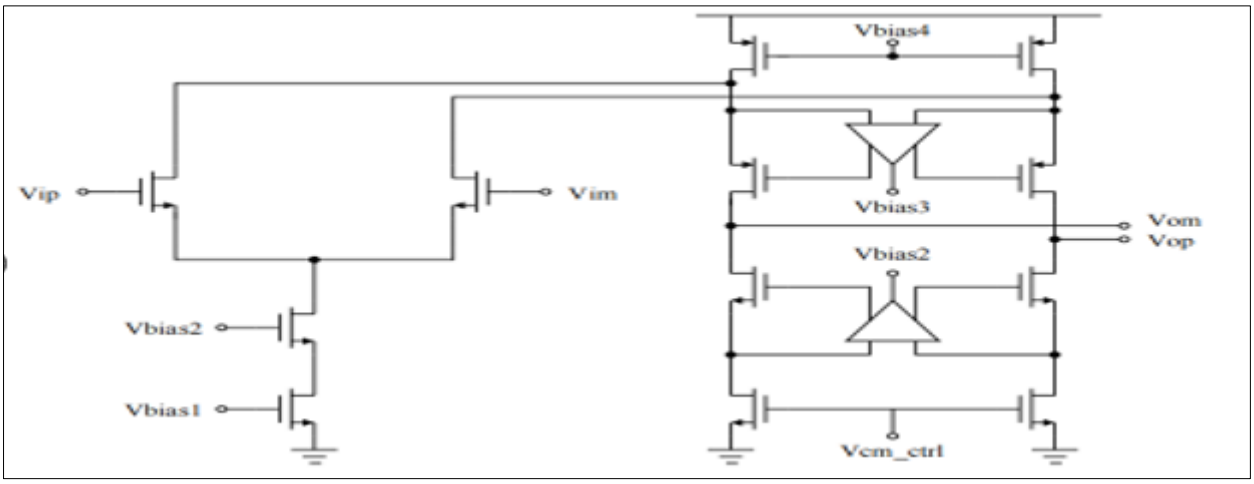


Figure 55 Top Level Opamp Circuit

The boost amplifiers were designed using a pair of folded-cascode amplifiers with opposite polarities. Specifically, the top amplifier utilized an NMOS input pair (as depicted in Fig. 4.9), while the bottom amplifier employed a PMOS input pair. The inclusion of this extra cascode stage was made possible due to the need for near-zero output swing in the boost amplifiers. Furthermore, these amplifiers operated with minimal current requirements, primarily because their output load consisted solely of the gate capacitance associated with the cascode devices in the primary amplifier.

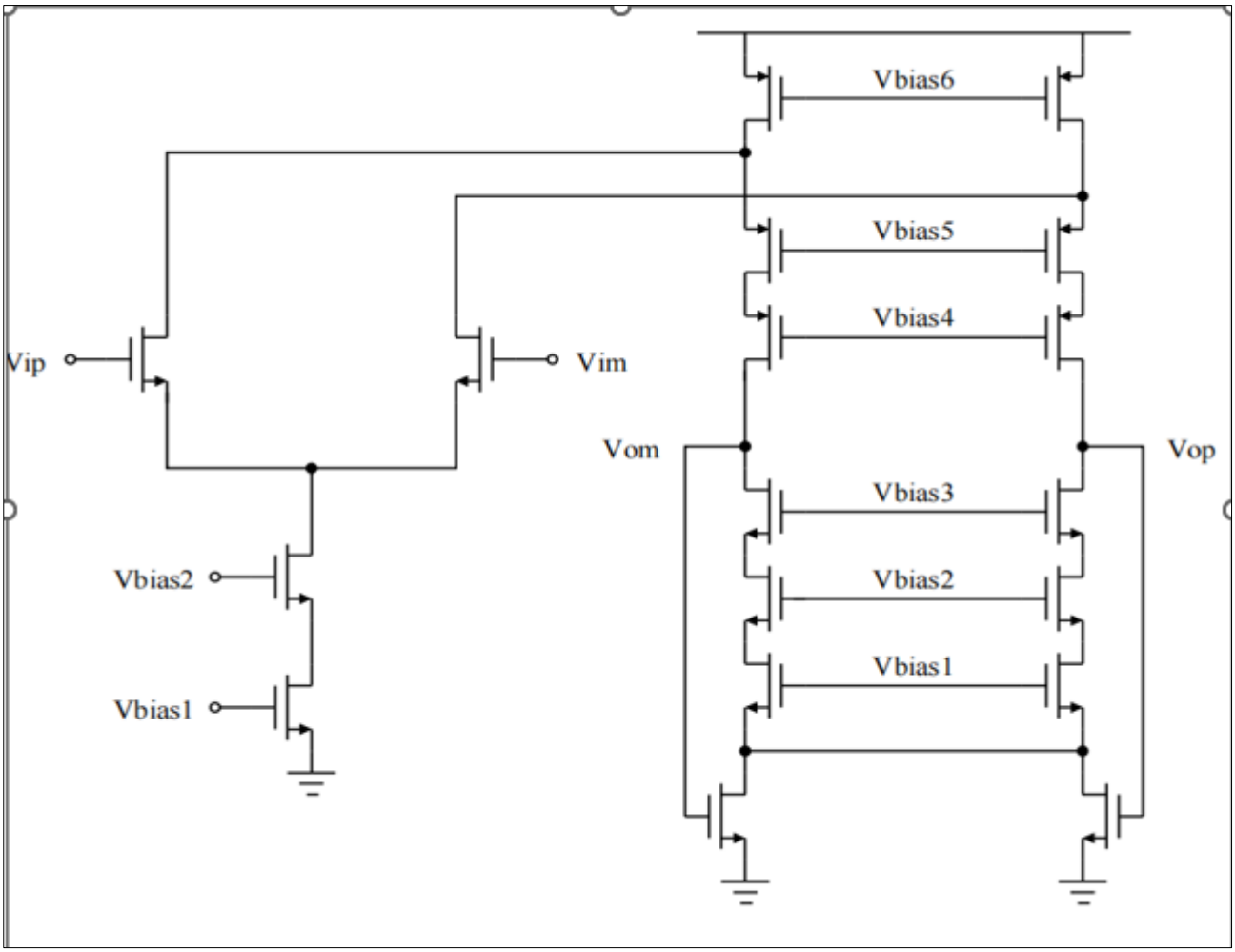


Figure 56 Gain-Boosting Amplifier (TOP)

4.2.4 Amplifier Linearity

Ensuring the output linearity of the amplifier is a crucial specification that must be attained. The amplifier achieves linearity primarily through its open-loop gain. The portion of the gain integrated into the feedback loop effectively mitigates the non-linear aspects of the amplifier's gain. However, it's imperative to maintain sufficiently high amplifier gain across the entire expected output range. Failing to do so would compromise the linearity of signals encompassing the full output spectrum. A common method for extrapolating the linearity of the amplifier system involves examining the open-loop gain of the amplifier across its complete output range. If the open-loop gain at the output remains above the desired input linearity throughout the entire output span, then the system should conform to the prescribed linearity requirements. This evaluation was conducted for the designed amplifier, and a graphical representation of gain versus output range is presented in Figure 57.

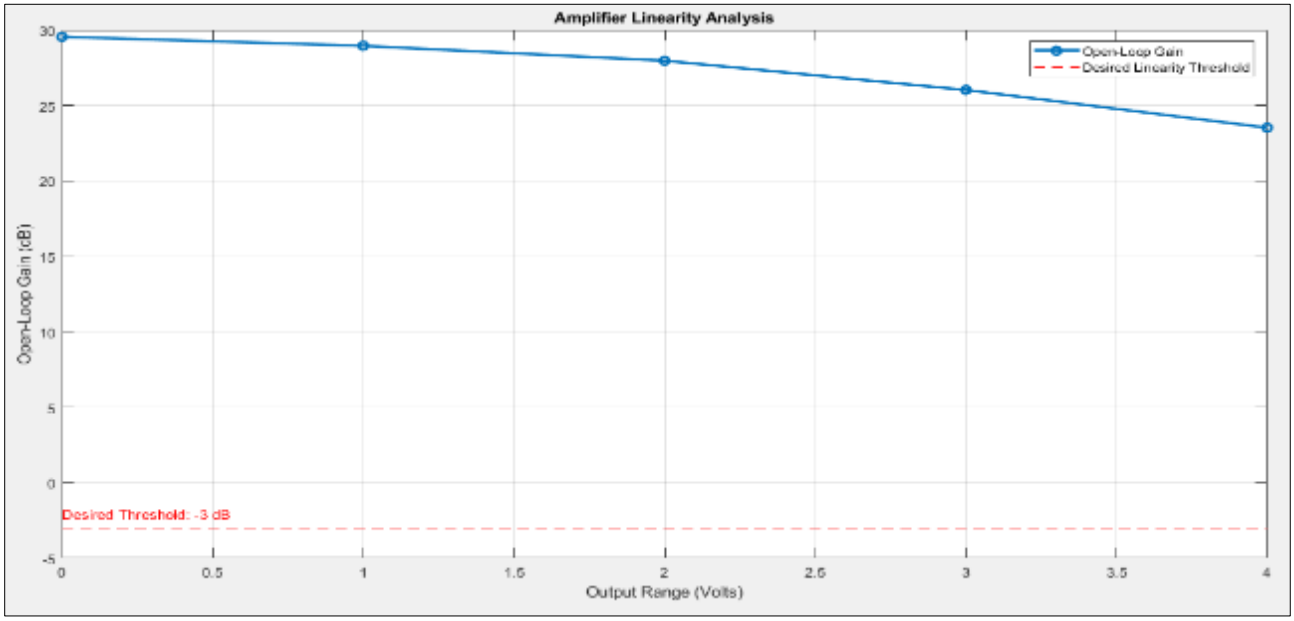


Figure 57 Open loop DC Gain over Output Range [Matlab**]**

A more rigorous approach for evaluating the linearity of the system, which incorporates the newly designed amplifier, involves the extraction of the amplifier's transfer function encompassing both linear and non-linear elements. This transfer function is then subjected to an input sine wave. The resulting waveform is scrutinized to identify any non-linear components, thereby allowing us to assess the amplifier's output linearity. This procedure was executed for the designed amplifier. The transfer function was meticulously extrapolated across multiple data points (as depicted in Figure 58 and subsequently imported into MATLAB. Employing a polynomial function fit, the imported curve was approximated, enabling the application of a sine wave based on the resulting equation. Subsequently, the non-linear constituents were isolated through Fast Fourier Transform (FFT) analysis of the output, yielding a Total Harmonic Distortion (THD) value spanning a range of output signal amplitudes. The outcomes of this comprehensive analysis are presented in Figure 59. These findings convincingly demonstrate that the THD, when normalized by the open-loop gain, comfortably surpasses the minimum requirement for achieving a linearity exceeding 12 bits.

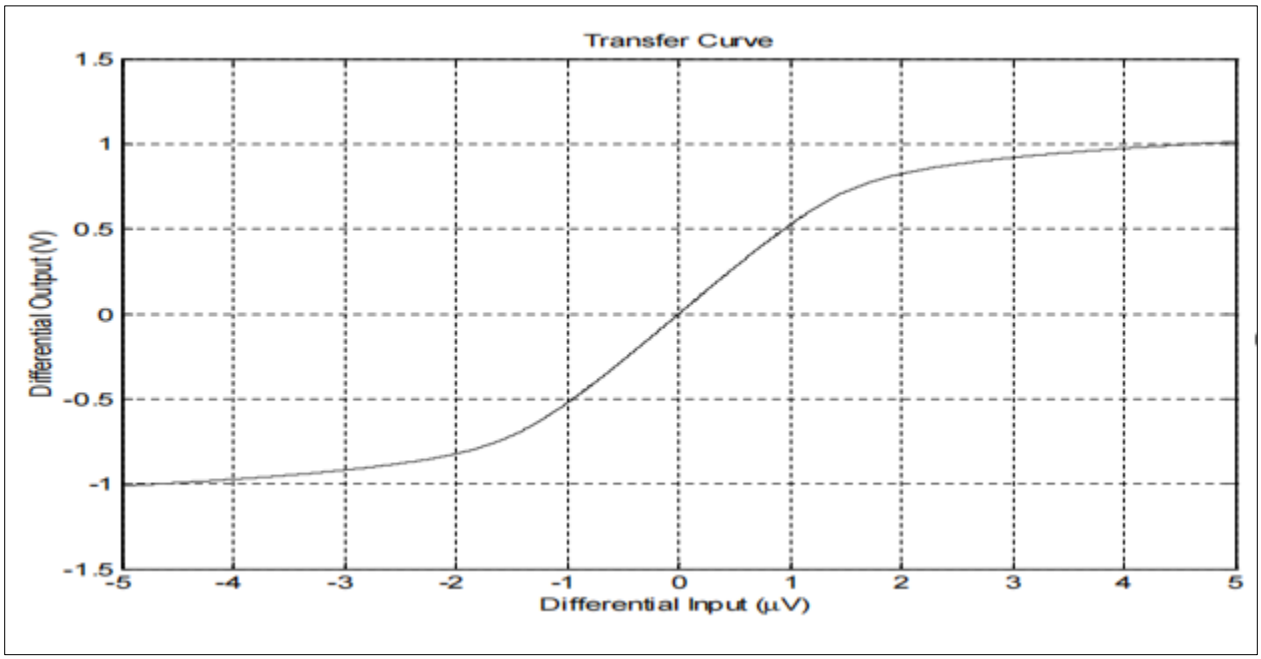


Figure 58 Amplifier DC Gain Curve

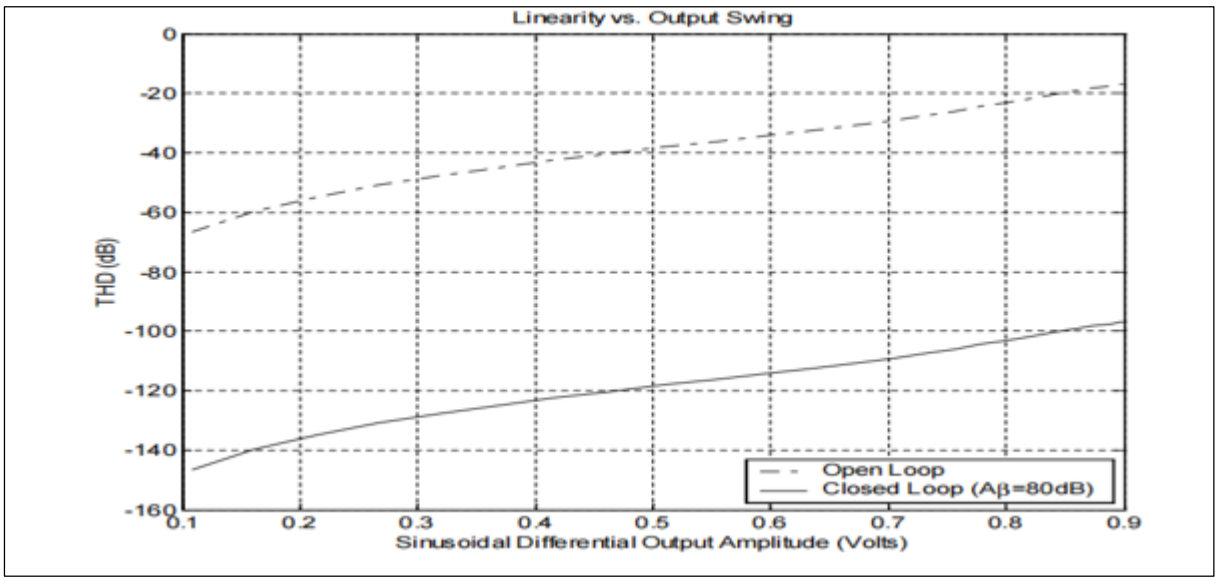


Figure 59 Total Harmonic Distortion (THD) vs Output Range

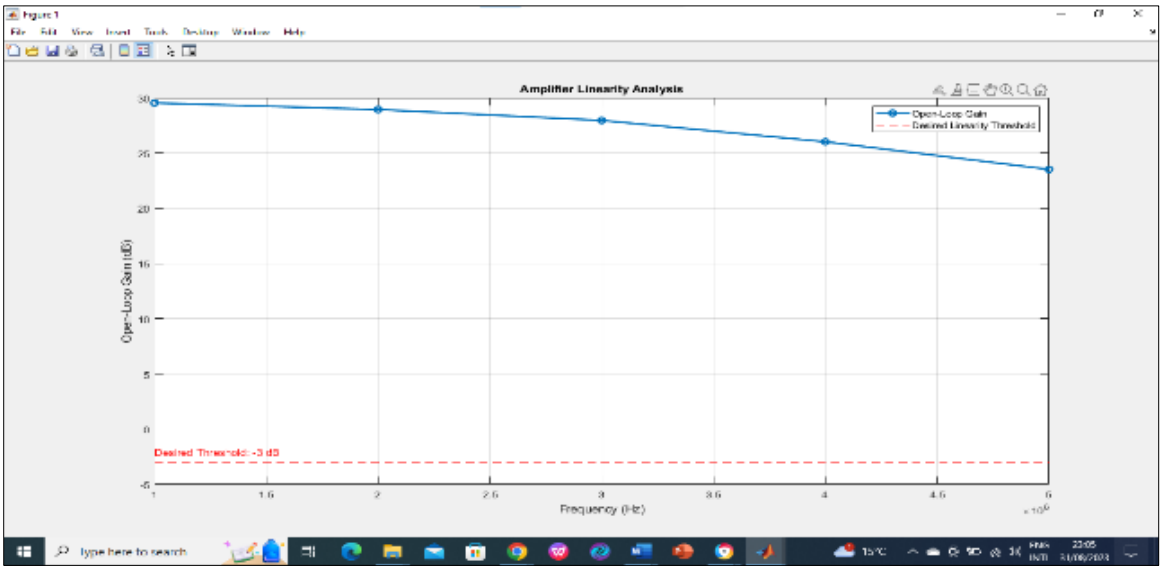


Figure 60 Amplifier Linearity Analysis[Matlab Code]

A transient analysis was conducted using SPECTRE to assess the linearity of the output signal at maximum swing, employing an input frequency of 11 MHz. It's worth noting that the input frequency utilized in the MATLAB simulation can be considered DC since the transfer curve represents an extracted DC gain curve. The findings reveal exceptional linearity exceeding 13 bits at the output. Furthermore, an examination of the transient signal within this configuration reaffirms a settling accuracy of approximately 14 bits.

4.3 Completed MDAC configuration

The ultimate MDAC setup is illustrated in Figure 61. Here, the input is differentially sampled across sixteen unit-sized capacitors, with eight capacitors on each side. The middle terminal is maintained at the common mode of the input signal. Additionally, four unit-sized capacitors are allocated on each side for feedback purposes, although it's worth noting that two of them are exclusively employed for the radix calibration β extraction operation.

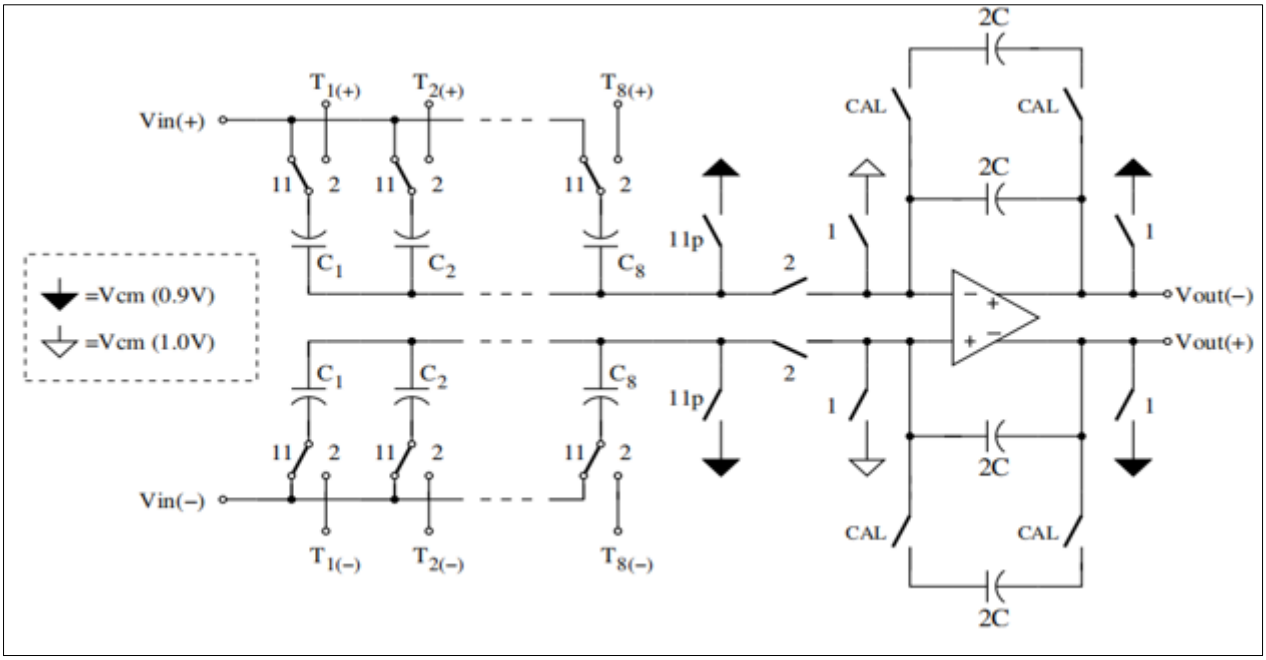


Figure 61 Final MDAC Implementation

The MDAC operates in the following manner. During its initial phase, the input signal is sampled across sixteen input capacitors, commonly referred to as the sampling phase.

In this phase, the operational amplifier (op-amp) is decoupled from the sampling process, with its inputs and outputs being momentarily shorted together. Simultaneously, within the primary amplifier, the switched-capacitor common-mode feedback is refreshed. Additionally, the four non-calibration measurement feedback capacitors are pre-charged to maintain the input of the amplifier at a consistent level, preventing any glitches when the MDAC transitions to the subsequent phase. Before concluding the first phase, a comparator resolves the bits from the initial stage and generates a reference code for use in the subsequent phase of MDAC operation. In the subsequent phase, known as the amplification phase, the thermometer DAC codes are applied to the capacitor reference selection switches. This action causes a controlled amount of charge to be added or subtracted from the input signal, allowing only the unresolved signal component to advance as charge to the feedback capacitors. The voltage gain experienced by the signal is determined by the capacitor ratio.

The MDAC's functionality is mathematically described as:

$$V_{out} = 4 \cdot V_{in} - d_i \cdot V_{REF2}, \tag{4.3}$$

where d_i is determined by the comparator output and can experience the values {-8/2, -7/2, -6/2, -5/2, -4/2, -3/2, -2/2, -1/2, 0, 1/2, 2/2, 3/2, 4/2, 5/2, 6/2, 7/2, 8/2}.

4.4 Comparator design

The initial stage comparator requires precise timing for continuous-time sampling in the Pipeline ADC architecture. An offset correction scheme, leveraging digital redundancy, is implemented for extensive sample mismatch correction. In Phase 1, one input capacitor samples the input signal, and the reference voltage is sampled on the other input capacitor. The comparator sampling circuit matches the MDAC sampling circuit but is 1/64th its size. The pre-amplifier's offset is also sampled. In Phase 2, the stored charges are averaged and applied to the pre-amplifier's inputs, amplified, and routed to the differential latch. After a brief delay, the output is available to digital circuits, initiating the MDAC amplification phase.

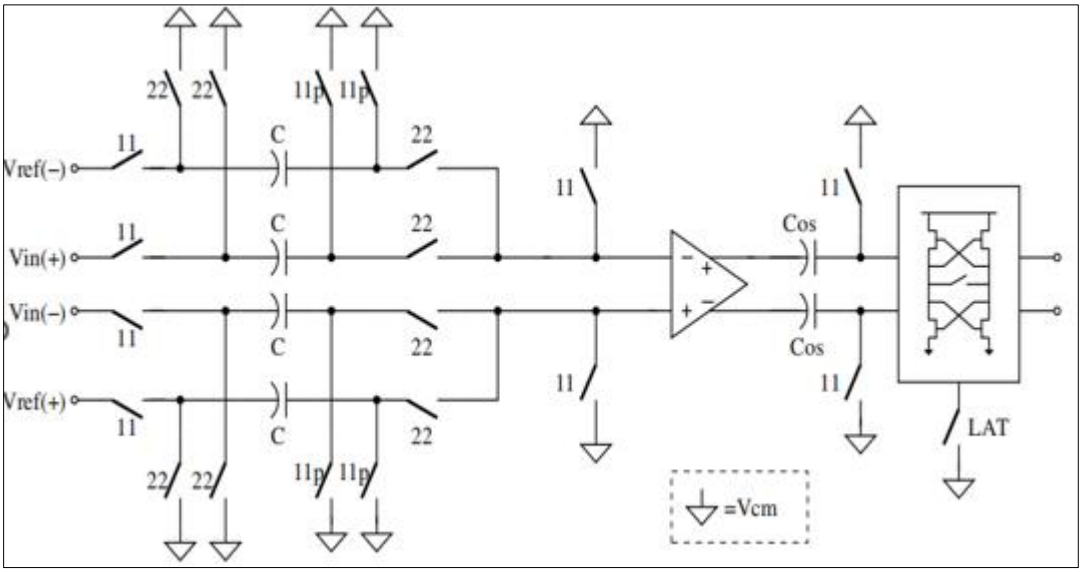


Figure 62 Top-Level Comparator Circuit

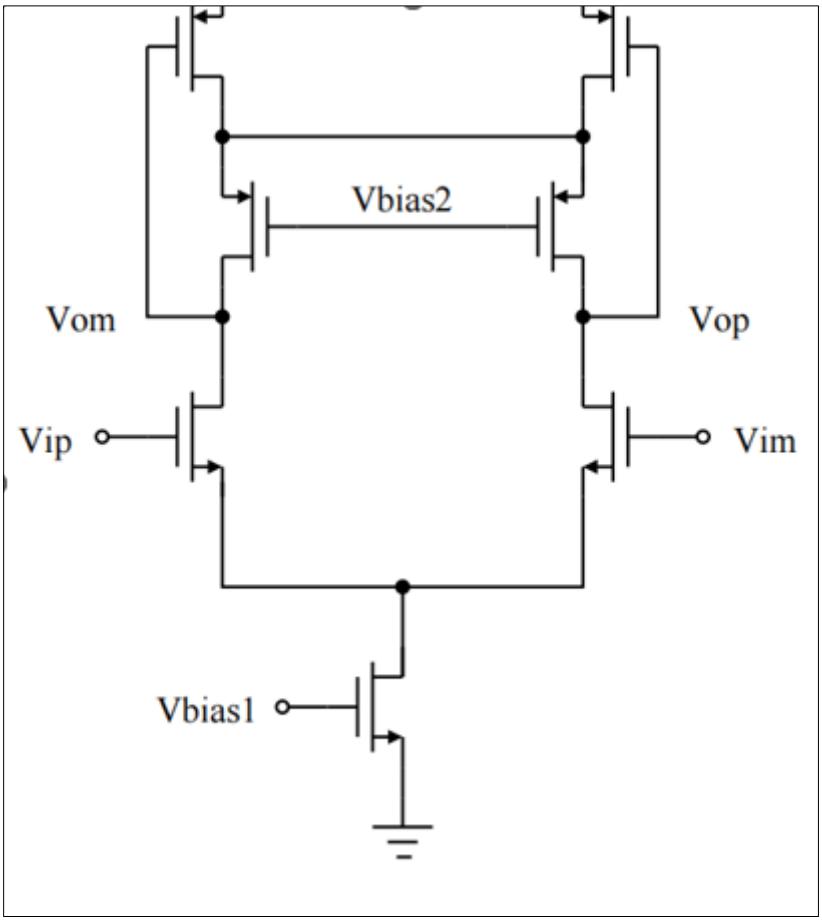


Figure 63 Comparator Pre-Amplifier

The simulation of the comparator confirms its functionality. It can accurately discern comparison signals as small as 10mV, which is less than 0.1 LSB (Least Significant Bit), within a rapid 4nS timeframe (comprising 2nS for the pre-amplification stage and an additional 2nS for latch regeneration).

Now, let's delve into the details of the pre-amplifier. The pre-amplifier is designed as a straightforward single-stage amplifier featuring low current consumption and moderate gain, as illustrated in Figure 64 and 65. It utilizes a linear operational MOSFET feedback mechanism for common-mode feedback. While this feedback approach might exhibit suboptimal performance across a wide input range, it is worth noting that this limitation holds minimal significance since the crucial comparator signals primarily consist of well-behaved, nearly zero-differential signals.

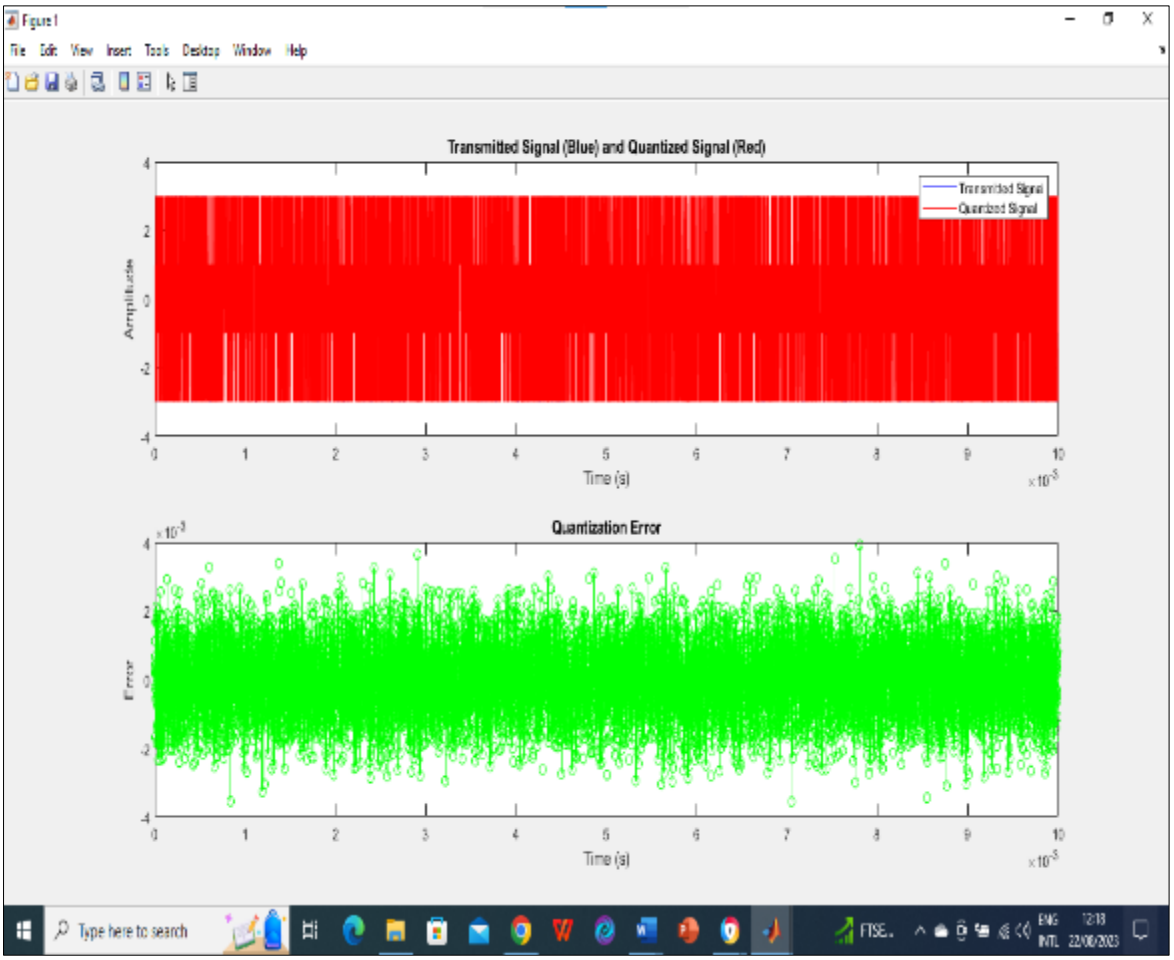


Figure 64 Amplifier and a comparator

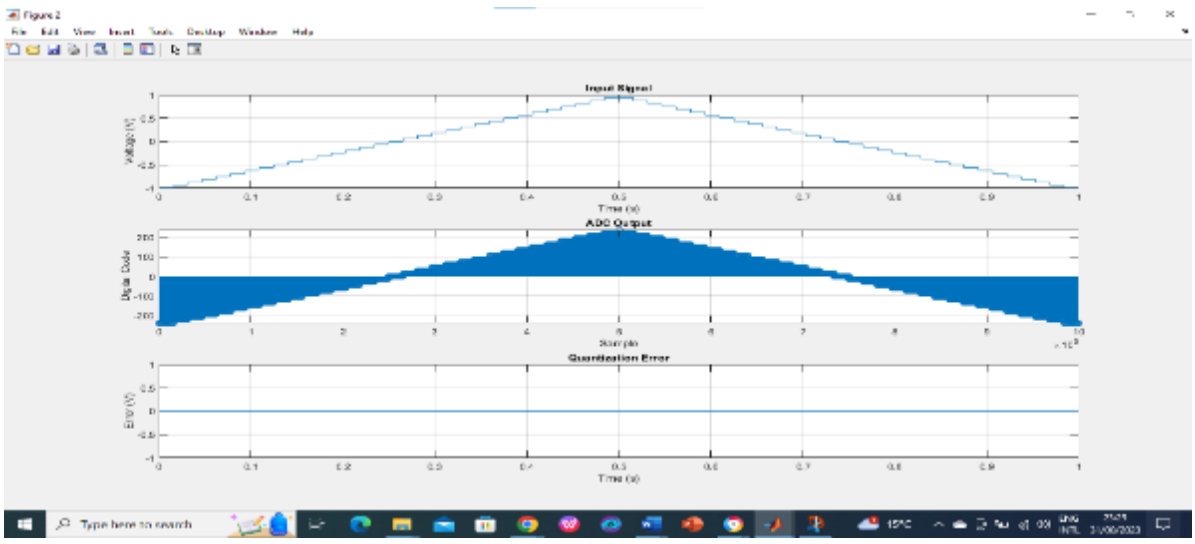


Figure 65 Amplifier and a comparator Gain.

5. Conclusion

This thesis has extensively explored the implementation of a high-resolution Pipeline ADC in a low-voltage process, developing and simulating a prototype incorporating several key solutions.

5.1 Key findings

- Sample-and-Hold Removal: Eliminating the Sample-and-Hold circuit reduces power consumption and area utilization, though it increases the complexity of the initial pipeline stage.
- Rail-to-Rail Input: Achieving a rail-to-rail input extends the signal range, enhancing the SNR. The MDAC operation accommodates two reference voltages, with a calibration scheme for reference mismatch correction.
- **Stage Size/Scaling Optimization**: An optimization technique determines the optimal number of bits per pipeline stage, improving noise and area performance.
- **Design Implementation**: Key aspects include a sampling structure with bootstrapped switches, a high-gain amplifier, high-accuracy comparators, digital support circuits for data latching and calibration, and layout work demonstrating feasibility.

5.2 Future directions

- **Alternative MDAC Structure**: Exploring single-reference VREF2 MDAC for rail-to-rail input without calibration could be valuable.
- **Stage Scaling Optimization**: Investigating methods for independent stage resolution and scaling adjustments may yield optimal configurations, along with better modeling of comparator stage power consumption.
- **Capacitor Matching**: Incorporating capacitor error correction mechanisms into the design could address matching issues, even with common-centroid arrays.

Compliance with ethical standards

Disclosure of Conflict of interest

No conflicts of interest to be disclosed.

References

[1] American National Standards Institute. Very-high-speed digital subscriber lines. System Requirements Document (T1E1.4/98-043R3); 1998.

[2] Siragusa E, Galton I. A digitally enhanced 1.8V 15b 40MS/s CMOS pipelined ADC. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 452–453.

[3] Liu H, Lee Z, Wu J. A 15b 20MS/s CMOS pipelined ADC with digital background calibration. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 454–455.

[4] Nair K, Harjani R. A 96dB SFDR 50MS/s digitally enhanced CMOS pipeline A/D converter. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 456–457.

[5] Chiu Y, Gray P, Nikolic B. A 1.8V 14b 10MS/s pipelined ADC in 0.18μm CMOS with 99dB SFDR. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 458–459.

[6] Mercer D. A 14-b 2.5 MSPS pipelined ADC with on-chip EPROM. IEEE Journal of Solid-State Circuits. 1996;31(1):70–76.

[7] Erdogan O, Hurst P, Lewis S. A 12-b digital-background-calibrated algorithmic ADC with −90-dB THD. IEEE Journal of Solid-State Circuits. 1999;34(12):1812–1820.

[8] Silva J. High-Performance Delta-Sigma Analog-to-Digital Converters [Ph.D. thesis]. Oregon State University, School of Electrical Engineering and Computer Science; 2004.

[9] Balmelli P, Huang Q. A 25 MS/s 14b 200mW ΣΔ modulator in 0.18μm CMOS. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 74–75.

[10] Putter B. ΣΔ ADC with finite impulse response feedback DAC. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 76–77.

[11] Semiconductor Industry Association. International technology roadmap for semiconductors; 2001. Available from: http://www.sia-online.org.

[12] Mehr I, Singer L. A 55-mW, 10-bit, 40-Msample/s nyquist-rate CMOS ADC. IEEE Journal of Solid-State Circuits. 2000;35(3):318–325.

[13] Chang D. PAPER TITLE UNKNOWN. IEEE Transaction on Circuits and Systems–I. 2005.

[14] Peterson J. A monolithic video A/D converter. IEEE Journal of Solid-State Circuits. 1979;SC-14(6):932–937.

[15] Song B, Lee S, Tompsett M. A 10-b 15-MHz CMOS recycling two-step A/D converter. IEEE Journal of Solid-State Circuits. 1990;25(6):1328–1338.

[16] Johns D, Martin K. Analog Integrated Circuit Design. John Wiley and Sons, Inc.; 1997.

[17] Razavi B, Wooley B. Design techniques for high-speed, high-resolution comparators. IEEE Journal of Solid-State Circuits. 1992;27(12):1916–1926.

[18] Lewis S, Gray P. A pipelined 5-Msample/s 9-bit analog-to-digital converter. IEEE Journal of Solid-State Circuits. 1987;SC-22(6):954–961.

[19] Lewis S, Fetterman H, Gross G, Ramachandran R, Viswanathan T. A 10-b 20-Msample/s analog-to-digital converter. IEEE Journal of Solid-State Circuits. 1992;27(3):351–358.

[20] Lee SH, Song BS. A direct code error calibration technique for two-step flash A/D converters. IEEE Journal of Solid-State Circuits. 1989;36:919–922.

[21] Lee S, Song B. Digital-domain calibration of multistep analog-to-digital converters. IEEE Journal of Solid-State Circuits. 1992;27(12):1679–1688.

[22] Lin Y, Kim B, Gray P. A 13-b 2.5-MHz self-calibrated pipelined A/D converter in 3μm CMOS. IEEE Journal of Solid-State Circuits. 1991;26(4):628–636.

[23] Temes G, Huang Y, Ferguson P. Switched-capacitor circuits with reduced sensitivity to amplifier gain. IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing. 1995;42(8):559–561.

[24] Song B, Tompsett M, Lakshmikumar K. A 12-bit 1-Msample/s capacitor error-averaging pipelined A/D converter. IEEE Journal of Solid-State Circuits. 1988;23(6):1324–1333.

[25] Van De Plassche R. Dynamic element matching for high-accuracy monolithic D/A converters. IEEE Journal of Solid-State Circuits. 1976;SC-11(6):795–800.

[26] Chen H, Song B, Bacrania K. A 14-b 20-MSample/s CMOS pipelined ADC. IEEE Journal of Solid-State Circuits. 2001;36(6):997–1001.

[27] Hastings A. The Art of Analog Layout. Prentice-Hall, Inc.; 2001.

[28] Li J, Moon U. A 1.8-V 67-mW 100-MS/s pipelined ADC using time-shifted CDS technique. IEEE Journal of Solid-State Circuits. 2004;39(9):1468–1476.

[29] Dessouky M, Kaiser A. Very low-voltage digital-audio ΣΔ modulator with 88-dB dynamic range using local switch bootstrapping. IEEE Journal of Solid-State Circuits. 2001;36(3):349–355.

[30] Abo A, Gray P. A 1.5-V, 10-bit, 14.3-MS/s CMOS pipeline analog-to-digital converter. IEEE Journal of Solid-State Circuits. 1999;34(5):599–606.

[31] Poujois R, et al. Low-level MOS transistor amplifier using storage techniques. IEEE International Solid-State Circuits Conference, Digest of Technical Papers. 1973; p. 152–153.

[32] Nagaraj K, Viswanathan T, Singhal K, Vlach J. Switched-capacitor circuits with reduced sensitivity to amplifier gain. IEEE Transactions on Circuits and Systems. 1987;CAS-34(5):571–574.

[33] Chiu Y. Inherently linear capacitor error-averaging techniques for pipelined A/D conversion. IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing. 2000;47(3):229–232.

[34] Ryu S, Ray S, Song B, Cho G, Bacrania K. A 14b-linear capacitor self-trimming pipelined ADC. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 464–465.

[35] Fu D, Dyer K, Lewis S, Hurst P. A digital background calibration technique for time-interleaved analog-to-digital converters. IEEE Journal of Solid-State Circuits. 1998;33(12):1904–1911.

[36] Ming J, Lewis S. An 8-bit 80-Msample/s pipelined analog-to-digital converter with background calibration. IEEE Journal of Solid-State Circuits. 2001;36(12):1489–1497.

[37] Karanicolas A, Lee H, Bacrania K. A 15-b 1-Msample/s digitally self-calibrated pipeline ADC. IEEE Journal of Solid-State Circuits. 1993;28(12):1207–1215.

[38] Murmann B, Boser B. A 12-bit 75MS/s pipelined ADC using open-loop residue amplification. IEEE Journal of Solid-State Circuits. 2003;38(12):2040–2050.

[39] Grace C, Hurst P, Lewis S. A 12b 80MS/s pipelined ADC with bootstrapped digital calibration. In: IEEE International Solid-State Circuits Conference, Digest of Technical Papers, February 2004. Vol. 1, p. 460–461.

[40] Yang W, Kelly D, Mehr I, Sayuk M, Singer L. A 3-V 340-mW 14-b 75-Msample/s CMOS ADC with 85-dB SFDR at nyquist input. IEEE Journal of Solid-State Circuits. 2001;36(12):1931–1936.

[41] Shyu J, Temes G, Yao K. Random errors in MOS capacitors. IEEE Journal of Solid-State Circuits. 1982;SC-17(6):1070–1076

[42] Prasad R, Kamal S, Sharma PK, Oelmüller R, Varma A. Root endophyte Piriformospora indica DSM 11827 alters plant morphology, enhances biomass and antioxidant activity of medicinal plant *Bacopa monniera*. Journal of basic microbiology. 2013 Dec; 53(12):1016-24. **(example of Journal article)**

Appendix

CMOS SWITCH SAMPLING LINEARITY

```
% Define parameters

bit_resolution = 14; % Desired ENOB of the ADC

mosfet_W = 1e-6;    % MOSFET channel width in meters

mosfet_L = 180e-9;  % MOSFET channel length in meters

VDD = 1.8;          % Positive power supply voltage in volts

VSS = -1.8;         % Negative power supply voltage in volts

VTH = 0.2;          % Threshold voltage in volts

% Create an array of input signal voltages

input_signal = linspace(VSS, VDD, 1000);

% Initialize arrays to store DNL values

DNL = zeros(size(input_signal));

% Loop through each input signal value

for i = 1:length(input_signal)

    Vsignal = input_signal(i); % Input signal voltage

    VGS = Vsignal + VTH;      % Gate-source voltage

    % Calculate the signal-dependent resistance

    unCox = 50e-6;            % Device constant for a typical MOSFET

    signal_dependent_resistance = (unCox * mosfet_W / mosfet_L) * (VGS - VTH);

    % Calculate the DNL

    ENOB = log2((2^bit_resolution - 1) / (2 * signal_dependent_resistance));
```

```
    DNL(i) = ENOB - bit_resolution;
end
% Plot the DNL graph
figure;
plot(input_signal, DNL);
xlabel('Input Signal Voltage (V)');
ylabel('DNL (bits)');
title('Differential Non-Linearity (DNL) vs. Input Signal Voltage');
grid on;
```

AMPLIFIER LINEARITY ANALYSIS

```
% Amplifier Linearity Analysis
% Load the transfer function or frequency response data of the amplifier
% Replace 'frequency' and 'gain' with your actual data
frequency = [1e6, 2e6, 3e6, 4e6, 5e6]; % Frequency values (Hz)
gain = [30, 28, 25, 20, 15]; % Gain values (dB)
% Define the desired input linearity threshold (e.g., -3 dB)
desiredLinearityThreshold = -3; % dB
% Calculate the open-loop gain at the output
openLoopGain = 20 * log10(gain);
% Plot the open-loop gain versus output range
figure;
plot(frequency, openLoopGain, '-o', 'LineWidth', 2);
xlabel('Frequency (Hz)');
ylabel('Open-Loop Gain (dB)');
title('Amplifier Linearity Analysis');
grid on;
% Add a horizontal line at the desired linearity threshold
hold on;
line([min(frequency), max(frequency)], [desiredLinearityThreshold,
desiredLinearityThreshold], 'Color', 'r', 'LineStyle', '--');
```

```
hold off;

% Add legend and annotations

legend('Open-Loop Gain', 'Desired Linearity Threshold');

text(min(frequency), desiredLinearityThreshold + 1, ['Desired Threshold: ', num2str(desiredLinearityThreshold), ' dB'], 'Color', 'r');

% Customize the plot as needed (e.g., axis limits, labels, etc.)

% xlim([min(frequency), max(frequency)]);

% ylim([min(openLoopGain) - 5, max(openLoopGain) + 5]);

% legend('Location', 'best');

% grid minor;

% Save the plot as an image (optional)

% saveas(gcf, 'Amplifier_Linearity_Analysis.png');

% Display the plot
```

OPEN LOOP DC GAIN OVER OUTPUT RANGE

```
% Amplifier Linearity Analysis

% Load the transfer function or frequency response data of the amplifier

% Replace 'outputRange' and 'gain' with your actual data

outputRange = [0, 1, 2, 3, 4]; % Output range (e.g., in volts)

gain = [30, 28, 25, 20, 15]; % Gain values (dB)

% Define the desired input linearity threshold (e.g., -3 dB)

desiredLinearityThreshold = -3; % dB

% Calculate the open-loop gain in dB

openLoopGain = 20 * log10(gain);

% Plot the open-loop gain versus output range

figure;

plot(outputRange, openLoopGain, '-o', 'LineWidth', 2);

xlabel('Output Range (Volts)');

ylabel('Open-Loop Gain (dB)');

title('Amplifier Linearity Analysis');

grid on;
```

```
% Add a horizontal line at the desired linearity threshold

hold on;

line([min(outputRange),            max(outputRange)],            [desiredLinearityThreshold,
desiredLinearityThreshold], 'Color', 'r', 'LineStyle', '--');

hold off;

% Add legend and annotations

legend('Open-Loop Gain', 'Desired Linearity Threshold');

text(min(outputRange),   desiredLinearityThreshold   +   1,   ['Desired   Threshold:   ',
num2str(desiredLinearityThreshold), ' dB'], 'Color', 'r');

% Customize the plot as needed (e.g., axis limits, labels, etc.)

% xlim([min(outputRange), max(outputRange)]);

% ylim([min(openLoopGain) - 5, max(openLoopGain) + 5]);

% legend('Location', 'best');

% grid minor;

% Save the plot as an image (optional)

% saveas(gcf, 'Amplifier_Linearity_Analysis.png');

% Display the plot
```