Massively Parallel Analog Front-End Array for Flexible CMOS-based Brain-Computer Interfaces

Arnau Diez Clos

MSc Thesis

TU Delft

Massively Parallel Analog Front-End Array for Flexible CMOS-based Brain-Computer Interfaces

by

Arnau Diez Clos

In partial fulfillment of the requirements for the degree of

Master of Science

In Electrical Engineering

at the Delft University of Technology,

To be defended publicly on Friday, September 6, 2024 at 10:30 AM.

Supervisors:	Dr. D.G Muratore,	Department of Microelectronics, TU Delft
	Dr. ir. G.L.E. Monna,	Phosphoenix
Thesis committee:	Dr. D.G Muratore,	Department of Microelectronics, TU Delft
	Dr. ir. K. Bult,	Department of Microelectronics, TU Delft
	Dr. ir. G.L.E. Monna,	Phosphoenix
Project duration:	September 4, 2023 – September 6, 2024	
Student number:	5859875	

TU Delft

Abstract

Brain-computer interfaces (BCIs) have the potential to revolutionize human-computer interaction and offer significant biomedical benefits by enabling applications such as closed-loop neuromodulation, mobility restoration for spinal cord injury patients, and therapies for neurodegenerative diseases. To fully leverage BCIs' capabilities, it is crucial to enhance spatial resolution and robustness by increasing the number of recording channels and expanding brain area coverage. However, current approaches often focus on maximizing the channel count within a single recording analog front-end (AFE), limiting scalability and coverage.

This thesis introduces a novel architecture for a low-power and compact AFE, designed for seamless integration into a massively parallel array of distributed single-channel AFEs dedicated to micro-electrocorticography (μECoG) recording. The proposed AFE features a DC-coupled, chopper-stabilized low-noise boxcar sampler (LNB), followed by a passive switched-capacitor low-pass filter (SC-LPF) and a single-slope (SS) analog-to-digital converter (ADC). The LNB enhances anti-aliasing by introducing notches at multiples of the sampling frequency and effectively reduces chopping ripple. An 8-bit quantization is performed using a continuous-time comparator synchronized with a globally distributed ramp across all channels of the array. The SS-ADC achieves a 12-bit resolution at the Nyquist rate (1 kSps) through oversampling at a rate of 16 times (OSR = 16).

Additionally, a novel electrode DC offset (EDO) cancellation loop prevents LNB saturation due to large EDO values, ensuring reliable performance in practical scenarios. Notably, this design eliminates the need for large AC coupling capacitors traditionally used for EDO cancellation, leading to a more compact design and improved scalability for high-density neural recording applications.

The proposed architecture is implemented at the transistor level in 40 nm CMOS technology and extensively validated through simulations. The AFE achieves exceptional area efficiency, with an estimated footprint of only 0.0028 mm^2 per channel. Moreover, the AFE achieves an input-referred noise (IRN) of 1.69 μVrms over 0.5-500 Hz and provides an EDO compensation exceeding 100 mV$_{\text{pp}}$. The design exhibits an input impedance of 71.4 MΩ and a common-mode rejection ratio (CMRR) of 80.94 dB. Additionally, the AFE achieves a signal-to-noise ratio (SNR) of 43.3 dB with a 1 mV$_{\text{pp}}$ input signal. These results indicate that the presented AFE architecture, and specifically the novel compensation scheme, represent a promising approach for neural recording applications.

Acknowledgments

I am deeply grateful for the opportunity to work on this challenging topic. Developing a novel system architecture for this thesis has been both intellectually stimulating and demanding. The task of creating a concept with the potential to enhance existing methods required perseverance and a thorough exploration of the subject. These challenges have fueled my passion for Brain-Computer Interfaces and have led to a truly rewarding experience

I would like to extend my sincere gratitude to everyone who has supported me on this journey. First and foremost, I want to thank my supervisor, Dr. Dante Muratore, for inspiring me to pursue a career in the field of Brain-Computer Interfaces and for allowing me to conduct my master's thesis within his research group. The freedom he has granted me to explore and follow my own ideas, has been instrumental in my growth as a researcher. Moreover, his invaluable guidance throughout each step of this project has significantly improved my skills as a mixed-signal IC designer.

I am also deeply grateful to my co-supervisor, Dr. it. Bert Monna, for welcoming me into his team at Phosphoenix. His constant support and willingness to engage in extensive discussions about various aspects of the design have profoundly deepened my understanding of fundamental engineering principles and underscored the importance of a systematic and structured design approach.

Dr. Xiaohua Huang is also deserving of recognition for his insightful suggestions and ideas during times of difficulty. His contributions have been crucial in advancing the project. Additionally, I appreciate the support of my colleagues on the 16th floor throughout this process.

I am also grateful to my friends, both back home and those I've made in Delft, for their motivation throughout this journey. In particular, to my friend Amar Kohabir for creating a fun and supportive atmosphere during this challenging year.

During this time, I have also been incredibly fortunate to have the unwavering support of my girlfriend, Laura. Her constant encouragement and belief in me have been a source of immense strength.

Lastly, I want to express my deepest gratitude to my parents for their unconditional trust and support throughout my academic journey and my entire life.

Arnau Diez Clos
Delft, August 2024

Contents

List of Figures

List of Tables

List of Abbreviations

ADC	Analog to Digital Converter
AFE	Analog Front End
ALS	Amyotrophic Lateral Sclerosis
ASIC	Analog Specific Integrated Circuit
BCI	Brain Computer Interface
BLC	Bi-level Compensation Loop
CDAC	Capacitive Digital to Analog Converter
CFN	Capacitive Feedback Network
CHS	Chopper Stabilization
CMFB	Common Mode Feedback
CMRR	Common Mode Rejection Ratio
CTC	Continuous Time Comparator
DAC	Digital to Analog Converter
DDC	Direct-to-Digital Converter
DFF	D Flip-Flop
DR	Dynamic Range
DSL	DC Servo Loop
ECoG	Electrocorticography
EDA	Electronic Design Automation
EDO	Electrode DC Offset
EEG	Electroencephalogram
ENOB	Effective Number Of Bits
FF	Flip-Flop
FIR	Finite Impulse Response
HPF	High Pass Filter
HVT	High-Threshold Voltage
IDAC	Current Digital to Analog Converter
IRN	Input Referred Noise
IRO	Input Referred Offset
LNA	Low Noise Amplifier
LPF	Low Pass Filter
OTA	Operational Trans-conductance Amplifier
RDAC	Resistive Digital to Analog Converter
SAR	Successive Approximation Register
SC	Switched Capacitor
SFDR	Spurious Free Dynamic Range
SNDR	Signal to Noise and Distortion Ratio
SNR	Signal to Noise Ratio
SS	Single Slope
TDC	Time to Digital Converter
TDM	Time Division Multiplexing
THD	Total Harmonic Distortion
TFF	T Flip-Flop
VTC	Voltage to Time Converter
WIS	Windowed Integrated Sampling
μ**ECoG**	Micro-electrocorticography

1

Introduction

1.1. Brain-Computer Interfaces

Brain-computer interfaces (BCI) have the potential to revolutionize the interaction between humans and computers and are one of the fastest expanding fields of scientific inquiry [1]–[5]. BCIs are divided into unidirectional and bidirectional interfaces, according to the direction of their action. Unidirectional BCIs record neural information or stimulate the neural system. On the other hand, bidirectional interfaces allow for an exchange of information in both directions [3]. For instance, bidirectional BCIs can be employed in closed-loop neuromodulation systems, where real-time feedback is used to control neurostimulation parameters [6]. Another application involves enabling individuals with spinal cord injuries to regain mobility through systems that record brain signals and stimulate the corresponding muscles in the extremities [7].

Brain-computer interfaces offer significant benefits across various biomedical applications, particularly for individuals with disabilities such as limb paralysis [8], [9] and blindness [10], [11]. Additionally, BCIs have proven effective in alleviating the symptoms of various neurodegenerative diseases, including epilepsy [1], amyotrophic lateral sclerosis (ALS) [12], Parkinson's disease [13], and Alzheimer's disease [14], [15]. Figure 1.1 illustrates the operation principle of a bidirectional BCI.

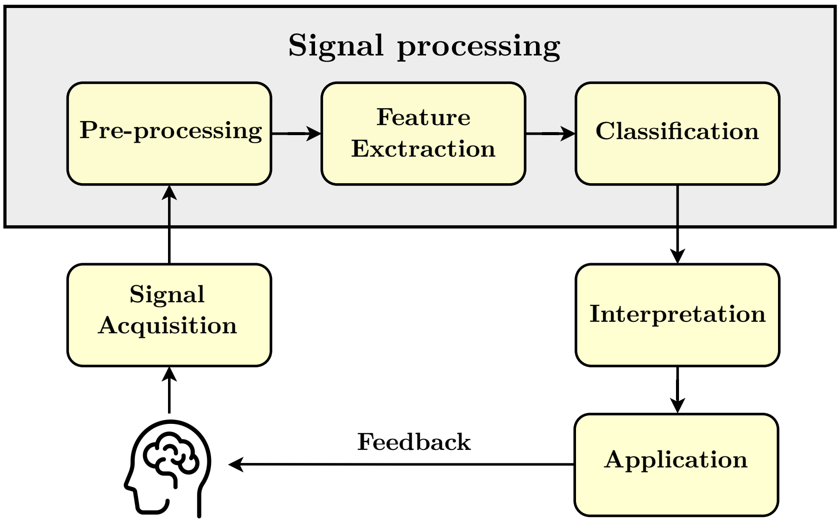


Figure 1.1: System level representation of a bidirectional BCI.

BCI systems obtain neural signals from the brain by recording electrodes through a spectrum of invasive and non-invasive techniques. The acquired signals are pre-processed to filter noise and artifacts to improve their signal-to-noise ratio (SNR). After that, processing algorithms are employed to extract the information of interest (features) from the preprocessed signals [16] (for instance, the intention of the user). Then, the obtained features go through a classification stage where multiple algorithms are used to determine the actual desired task (e.g. move the right leg forward). Ultimately, the obtained command is interpreted by its specific characteristics on the controlled device (e.g. electrically stimulating a specific point of the right quadriceps, to generate a desired leg movement).

1.1.1. Neural recording techniques

Within biological systems, including the human body, electrical signals are primarily mediated by the movement of ions, such as sodium, potassium, and chloride. This ionic conductivity presents a distinct challenge for interfacing these biological systems with electronic devices, whose function is based on the flow of electrons. Electrodes serve as a critical bridge, facilitating the conversion of ionic signals into a form interpretable by electronic instrumentation. The specific characteristics, placement strategies, and invasiveness of these electrodes define the various neural recording techniques employed. As depicted in Figure 1.2, these techniques provide a spectrum of options for neuroscience research and clinical applications.

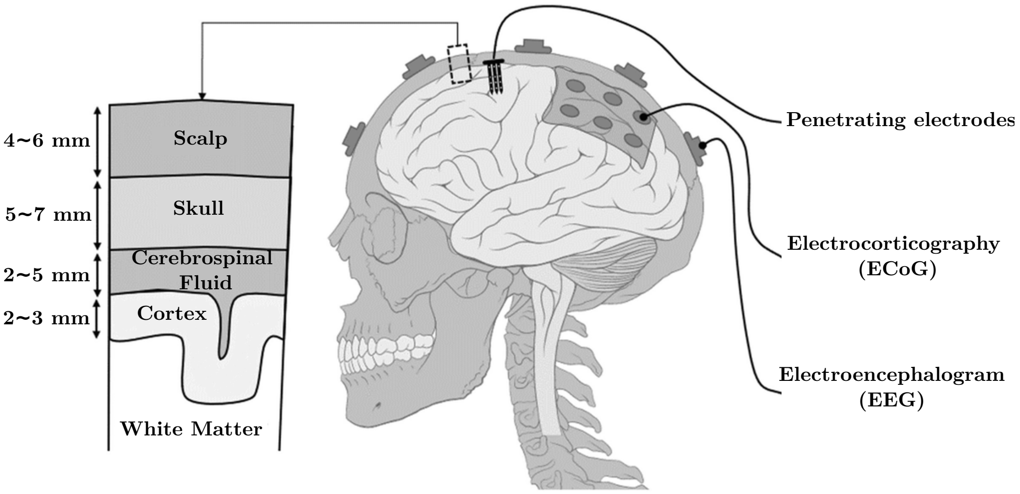


Figure 1.2: Representation of different neural recording techniques used in BCIs. Adapted from [17].

The Electroencephalogram (EEG) is a widely used non-invasive method for monitoring electrical brain activity [18], [19]. In EEG recording setups, the electrode arrays are placed at the scalp around 2 cm above the cortex, with each electrode recording the neural activity of approximately 4 cm [20]. This technique can cover large brain regions but suffers from poor spatial resolution and poor depth information [19]. On the other hand, neural recording using penetrating electrodes can provide unprecedented spatial and temporal resolutions [21], [22]. However, this approach has limited coverage and can't record from large brain regions. Additionally, most penetrating electrode arrays used in single-cell interfaces are not mechanically compatible with brain tissue. This incompatibility can lead to tissue scarring and trigger a biological foreign-body response, ultimately degrading the quality of long-term neural signal recordings [23]–[26].

Electrocorticography (ECoG) is a relatively less invasive technique that involves recording electrical signals from the cerebral cortex via electrodes implanted on the cortical surface [5], [27], [28]. This method strikes a balance between spatial resolution and invasiveness. While clinical ECoG electrodes are typically large (about 4 mm in diameter), there are also flexible and conformal micro-electrocorticography (μECoG) electrode arrays. The μECoG electrodes are less than 1 mm in diameter, offer an improved spatial resolution, and are a promising option to capture network-level information with neural circuit resolution [28], [29]. Additionally, these arrays closely conform to the cortical surface, providing stable mechanical and electrical contact and reducing the potential for tissue scarring [29], [30]. Therefore, it is the recording technique of focus for this thesis.

1.1.2. Neural signals of interest

μECoG electrodes primarily record local field potentials (LFPs) [28], which represent the aggregated electrical activity of small populations of neurons around the recording site. LFPs are being increasingly investigated in BCI setups, with performance comparable to those using single-resolution penetrating arrays [31]–[33]. These signals present an amplitude range of 10 μV_{pp} to 1 mV_{pp} and a band of interest from 0.5 to 500 Hz [28], [34].

1.1.3. Electrode-tissue interface

When the μECoG electrodes are implanted, a metal-electrolyte interface is formed and represented by an equivalent circuit seen in Figure 1.3 [35]–[37].

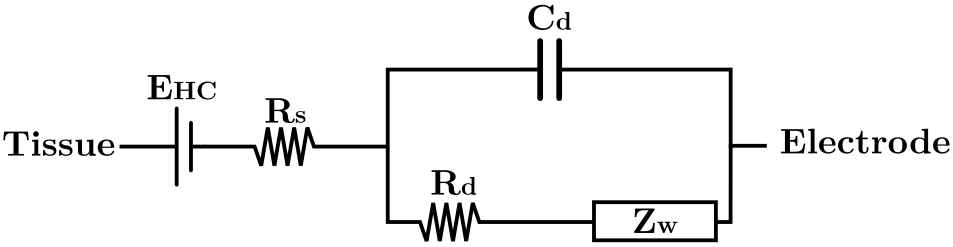


Figure 1.3: Representation of an electrode-tissue interface with non-polarizable faradaic μECoG electrodes.

Equivalent impedance

The conductivity of the electrolyte, represented by a spreading resistance (R_s), forms a connection between the electrode-electrolyte interface and the tissue. The conduction mechanisms between the electrolyte and the electrode include faradaic and capacitive conduction, along with diffusion. Faradaic conduction involves the transfer of electrons between the electrode and electrolyte (due to reduction and oxidation reactions) and exhibits ohmic characteristics (R_d). Inserting an electrode into an electrolyte also leads to the formation of a layer composed of oriented water dipoles and solvated ions surrounding the electrode [37], which creates a dielectric layer between the electrode and the electrolyte. This formation has a capacitive behavior (C_d), known as the double-layer capacitance. At lower frequencies, the conduction is purely resistive (R_d) and governed by faradaic processes. At higher frequencies the conduction becomes partly capacitive as Rd becomes bypassed by C_d [35].

Additionally, during electrochemical processes, ions must migrate through the electrolyte to reach the electrode surface. This migration is not instantaneous, and slower diffusion rates hinder the overall current flow. The Warburg impedance (Z_w) mathematically represents this diffusional hindrance. However, this element is typically negligible in practical experiments involving neural recording electrodes [28], [37].

Electrode DC Offset

The transfer of electrons results in positive ions gathering near the negatively charged electrode, and vice versa. This separation of charges creates an electric field, leading to a potential difference across the interface, known as the half-cell potential (E_{HC}) [37]. The potential difference between the recording and reference electrodes is known as electrode DC offset (EDO), which can vary with electrochemical reactions on the electrode's surface. The EDO with an amplitude of tens of mV [34], [38], [39] is recorded by the electronic devices along with the desired neural information.

Electrode thermal noise

The random thermal fluctuations of the charged particles within the electrode material result in a potential variation superimposed into the recorded neural signal, known as electrode thermal noise. The power spectral density (PSD) of the electrode thermal noise is exposed in Equation 1.1

$$\overline{v_n^2} = 4k_B T R_d \tag{1.1}$$

Where k_B is the Boltzmann constant (1.38×10^{-23} J/K) and T is the absolute temperature in Kelvin.

1.1.4. Analog front-end requirements

As mentioned, μECoG signals have a low amplitude and therefore are easily corrupted by noise. An analog front end (AFE) acts as the first stage in processing these signals. It amplifies the signal while minimizing unwanted noise, limits its bandwidth to avoid high-frequency interference, and converts the analog signal to the digital domain for subsequent BCI steps (shown in Figure 1.1). The main constraints and requirements are listed below.

Input impedance

A critical aspect of the design of an AFE is adapting its input impedance to the output impedance of the electrode. The ionic current i_c flows from the brain tissue and generates a potential V_{tissue} (Fig. 1.4). V_{tissue} appears at the input of the AFE through the electrode and hence the AFE input impedance (Z_{in}) forms a potential divider with the electrode impedance, leading to attenuation as described in Equation 1.2.

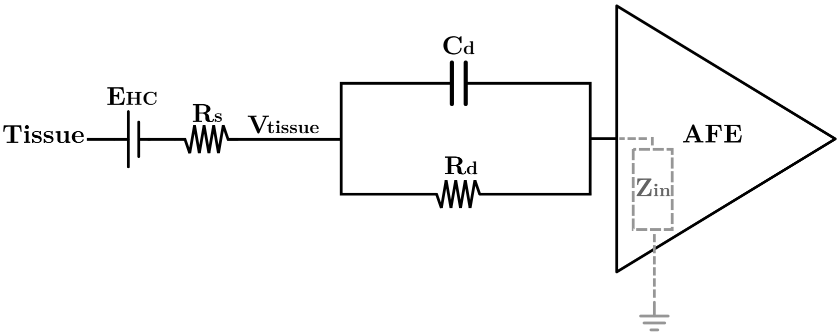


Figure 1.4: Schematic of a typical neural interface composed of recording electrodes and an AFE.

$$v_{in}(s) = v_{tissue} \cdot \frac{Z_{in}(s)}{Z_{in}(s) + R_d \| \frac{1}{sC_{dl}}} \tag{1.2}$$

The severity of the attenuation is highly dependent on the impedance of the electrode. Electrode impedance can vary significantly based on the electrode material and diameter. As reported in [29], [40], electrodes with diameters ranging from 50 μm to 300 μm can exhibit impedance values as high as 1.1 MΩ at 10 Hz, and as low as 10 kΩ at 10 kHz.

Noise performance

The AFE must be designed with a sufficiently low noise level (will be detailed in section 2.2) to preserve the integrity of the recorded neural signals. While there is no definitive standard for the optimal noise value, a noise level below 2 μV_{rms} is commonly targeted for μECoG recording setups [29], [34]. However, this value strongly depends on the chosen electrode as it introduces its own thermal noise. Therefore, the AFE design should account for this noise floor to avoid over-designing, which can lead to excessive power consumption [28], [35].

Power consumption

Power consumption is also another factor to consider. This limitation arises because the neural tissue is highly sensitive to heat, and exceeding a certain threshold can lead to damage or even necrosis [37], [41]. This thermal constraint typically translates to a maximum power dissipation of around 1 mW/mm^2. While this value represents the absolute limit, the actual power available to the AFE is often even lower due to the capacity of the implant's power source or link.

Common-mode rejection ratio

To ensure the integrity of neural signal recordings, it is crucial to mitigate common-mode interferences, such as the biological background noise (from the electrical activity of the surrounding tissue), the common-mode E_{HC} from both the recording and reference electrodes and crosstalk from surrounding recording channels. These interferences can corrupt the neural signals of interest. The common-mode rejection ratio (CMRR) is used to quantify the ability of the device to reject common-mode signals (eq. 1.3).

$$CMRR = 20\log_{10}(\frac{A_{dm}}{A_{cm}}) \tag{1.3}$$

Where A_{dm} is the differential gain of the AFE and A_{cm} is the common-mode gain. A neural recording AFE must have a high enough CMRR ($>$ 75 dB) to effectively reject the unwanted common mode interferences [42].

1.1.5. High density recording arrays

Increasing the number of recording channels in BCIs significantly enhances their functionality and effectiveness. More channels improve spatial resolution, allowing for a finer mapping of brain activity and a more precise interpretation of neural signals. In addition, it contributes to the robustness of the system by providing redundancy that helps counteract signal variability. With more detailed brain mapping capabilities, BCIs can better support advanced research into the brain's functional networks and aid in understanding and treating neurological disorders [43]. The adaptability of BCIs also improves, as they can be customized to better fit individual users and specific applications. Thus, future BCIs will simultaneously record from tens of thousands of electrodes, increasing the recording electronic's area and power efficiency requirement [44].

The design of high-density μECoG arrays is challenging, as each electrode needs to be addressed individually, resulting in a wiring bottleneck from the electrode array to the recording circuits [29]. However, flexible electronics [45]–[47], provide a solution to the routing problem, allowing for the integration of active circuits into the large area of polymeric substrate.

Multiplexing techniques, applied directly at the recording electrodes, are widely used to increase channel counts while sharing a single AFE, thus enhancing area efficiency and maintaining competitive power efficiency [48], [49]. However, this approach introduces challenges, such as increased noise folding from the required higher bandwidth and potentially degrading signal quality. Additionally, while high channel counts are achievable, it lacks the flexibility to allocate recording channels to specific zones. It also provides limited area coverage as the channels must be close to the recording AFE to avoid increased crosstalk, interference, and routing complexity.

Employing an array of very low-power and compact Analog Front Ends (AFEs) for each recording electrode can significantly enhance scalability by avoiding routing congestion and improving signal integrity, as demonstrated in [50]–[52]. In addition, these designs can offer greater area coverage and provide the flexibility to tailor channel distribution across different brain regions. However, their solutions are implemented in rigid silicon analog-specific integrated circuits (ASICs) or utilize multiple single-AFE ASICs embedded in flexible materials. Therefore, embedding high-density AFE arrays in a single flexible and conformable ASIC would further advance this trend.

In this proposed implementation, a centralized electronics hub manages wireless data transfer, power distribution, and communication with the spatially distributed AFEs embedded in the flexible substrate (Fig. 1.5). While each AFE is dedicated to a single recording electrode, all channels share a common reference electrode. This configuration leverages the benefits of distributed AFEs, enhancing the system's overall flexibility, scalability, and performance.

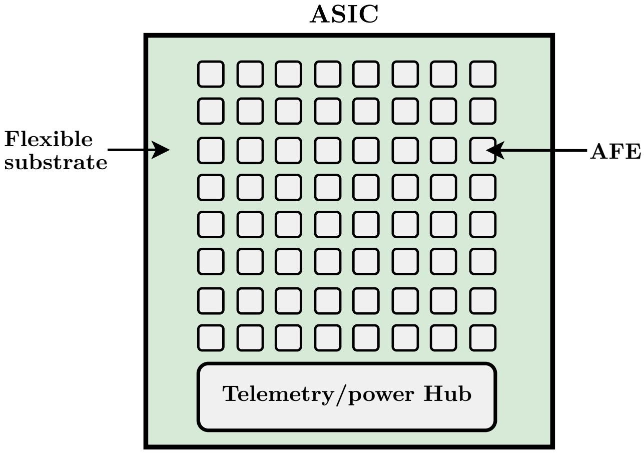


Figure 1.5: Array of single-channel recording AFE embedded into a flexible substrate (the pin pads and routing are not included in the representation).

1.2. Thesis and system requirements

This thesis focuses on developing a massively parallel array (scalable to thousands of channels) for the recording of μECoG signals. The complete system would require efforts in three key areas:

1. Design of an ultra-low-area, ultra-low-power single-channel analog front-end (AFE) for muECoG signal recording.
2. Development of a scalable array architecture for thousands of channels, with emphasis on signal routing and communication between AFEs and the central hub.
3. Development of a flexible and fully conformable application-specific integrated circuit (ASIC) using novel microfabrication techniques.

The scope of this thesis is limited to the first point: the design of a single-channel AFE for μECoG signal recording. The other two aspects, while crucial for the complete system, are considered future work and are not addressed in this study. Table 1.1 summarizes the system-level requirements for the AFE design.

1.3. Thesis organization

This thesis is organized as follows. Chapter 2 provides an overview of the prior art in neural AFEs. Chapter 3 presents the analytical foundation and the high-level design choices for the system-level design. Chapter 4 contains the circuit-level design of the neural AFE. Chapter 5 introduces the obtained simulation results. Chapter 6 concludes the thesis and offers a discussion about future work to further improve the presented results.

Table 1.1: Requirements for a neural AFE for the recording of μECoG signals.

Parameter	Value	Justification
Channels/AFE	1	Enhance scalability and area coverage [50]–[52].
Bandwidth	0.5 Hz - 500 Hz	Frequency range of the μECoG signals [28], [34]
Noise	$< 2\ \mu\text{V}_{rms}$	Differentiate neural signals from noise [28], [34]
EDO Rejection	$>100\ mV_{pp}$	Eliminate the EDO present at the input of the AFE [34], [38], [39]
Input impedance	>40 MΩ	Minimize attenuation of the recorded signal [29], [40]
Power density	< 1 mW/mm^2	Avoid tissue damage due to an increase in temperature [37], [41]
CMRR	>75 dB	Prevent the degradation of the neural signals due to common mode interferences [42]
Area/channel	$< 100\mu m \times 100\mu m$	Scalable to 1000+ channels and compatible size with most μECoG electrodes [28], [40]

2

Prior art

The system implementation for an AFE designed to record μECoG signals involves several critical components (Fig. 2.1). Firstly, a high-pass filter (HPF) is used to reject the EDO present in the recorded signal. After that, a low-noise amplifier (LNA) is employed to amplify the neural signals while minimizing noise addition. This is followed by a low-pass filter (LPF) to limit the signal bandwidth. Finally, an analog-to-digital converter (ADC) digitalizes the pre-amplified and filtered analog signals.

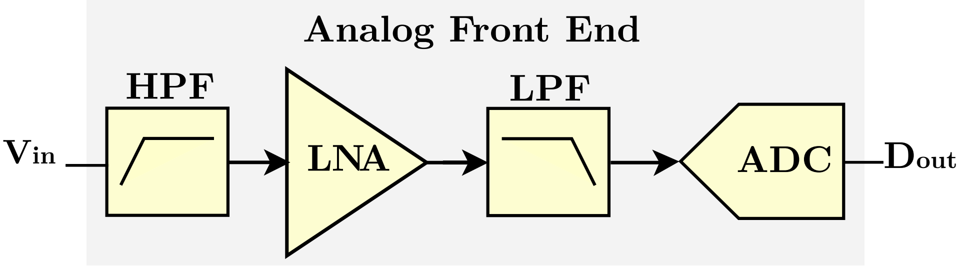


Figure 2.1: System level representation of a neural recording AFE.

This chapter investigates the design and implementation of AFEs for neural recording systems by examining various approaches found in prior art.

2.1. Amplifier and EDO rejection

Neural signals have a very small amplitude (as discussed in chapter 1), and need to be amplified before their digitalization by an ADC (Fig. 2.1). Additionally, the presence of the EDO (with orders of magnitude higher in amplitude) at the input can potentially saturate the amplifier. To accommodate both the neural signal and the EDO, a huge dynamic range (above 80 dB) would be required (Fig. 2.2).

Several techniques have been proposed to address this issue without requiring an unnecessarily large AFE dynamic range.

2.1.1. AC Coupling

AC-coupled amplifiers employ passive high-pass filtering with an RC network at the input of the LNA for rail-to-rail EDO rejection. Multiple variations of this approach can be distinguished.

AC-Coupled capacitive feedback network amplifier

A widely popular implementation is the AC-coupled capacitive feedback amplifier (CFN) [53]–[62]. In these designs, an operational transconductance amplifier (OTA) is used to amplify the signals. Figure 2.3 shows the system representation of a CFN configuration.

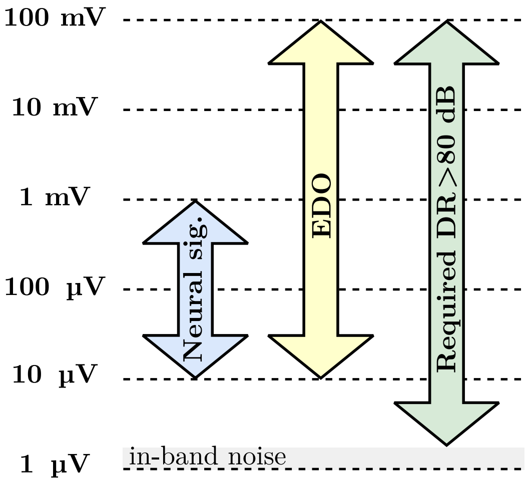


Figure 2.2: DR requirement of the recording circuit to record both EDO and neural signals.

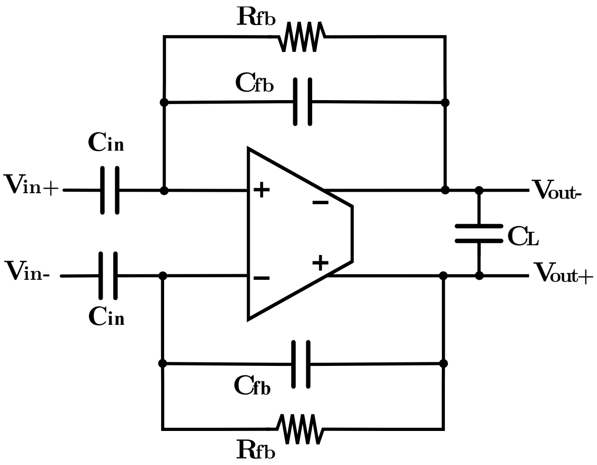


Figure 2.3: Circuit diagram of a capacitive feedback amplifier (CFN).

To block the EDO, a large capacitor is placed at the input (C_{in}). To set the HPF pole, the feedback capacitor (C_{fb}) is placed in parallel with a large resistive element (R_{fb}). The effective high-pass filter pole has a cut-off frequency of:

$$f_{HPF} = \frac{1}{2\pi \cdot R_{fb} \cdot C_{fb}} \tag{2.1}$$

The gain of this topology is derived in Equation 2.2.

$$\frac{V_{\text{out}}}{V_{\text{in}}} = \frac{C_{in}}{C_{fb}} \tag{2.2}$$

As can be seen, the gain is set with high accuracy because it only depends on the capacitive ratio. However, to set the high-pass filter (HPF) below the required 0.5 Hz, a very large feedback capacitor (C_{fb}) and/or feedback resistor (R_{fb}) are required (Eq. 2.1). Additionally, C_{fb} must be larger than the feedback parasitic capacitance in the OTA to establish an accurate pole, which then sets a constraint in its minimum size.

To achieve a high gain, C_{in} must be chosen with a large value relative to C_{fb} (according to Eq. 2.2). As a result, the input capacitors take up most of the area for each recording channel, limiting the scalability of this approach for high-density recording arrays (see Fig. 2.6).

To reduce the area of C_{in}, a T-Capacitor Network Topology (Fig. 2.4) can be used [38], [63], [64].

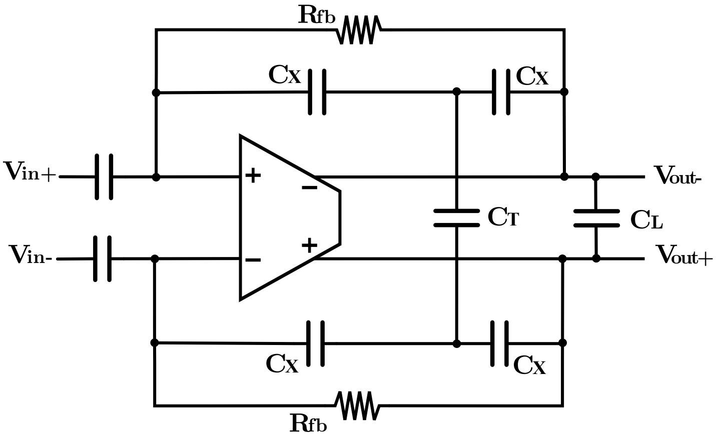


Figure 2.4: Circuit diagram of CFN with a T-capacitor network topology.

In this architecture, the feedback capacitor is replaced by a T-capacitor network, which reduces the total equivalent feedback capacitance, as shown in Equation 2.3.

$$C_{eq,fb} = \frac{C_X}{2\left(\frac{C_T}{C_X} + 1\right)} \tag{2.3}$$

This topology provides a lower requirement for the input capacitor but increases the minimum value of R_{fb} to set the HPF below 0.5 Hz, which leads to higher thermal noise contribution from the feedback resistor.

AC-coupled open-loop amplifiers

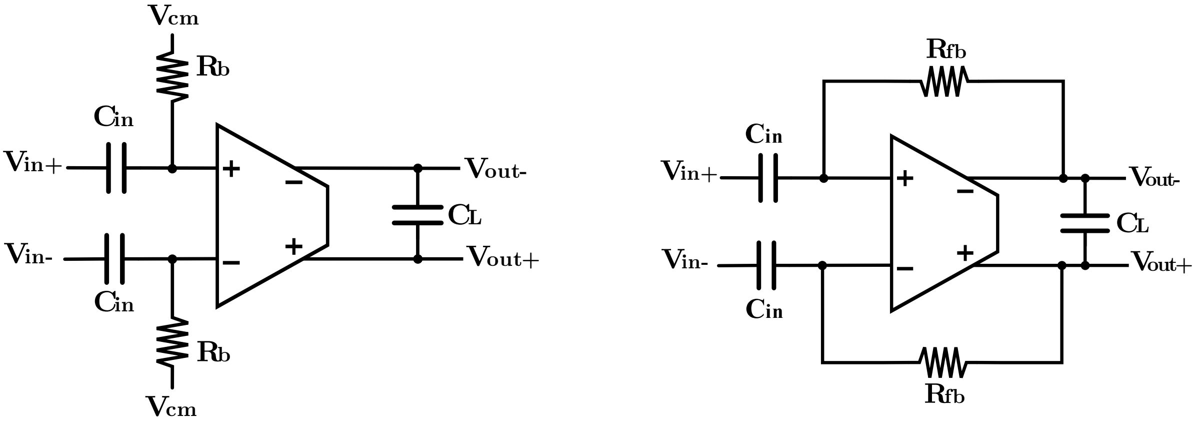
OTAs can operate without capacitive feedback in an open-loop configuration, as illustrated in Figure 2.5 [51], [65], [66]. In this setup, the high-pass filter (HPF) pole is established by the input capacitor (C_{in}) and the biasing resistor (R_b) (Fig. 2.5a) or by an auto-biasing resistor (R_{fb}) (Fig. 2.5b).

Since C_{in} no longer influences the amplifier's gain, the requirement for a minimum value is significantly lower, noticeably reducing the required area. Figure 2.6 shows an example of the layout for the open-loop and CFN amplifiers implemented in [67].

However, to set the HPF pole below 0.5 Hz, large input capacitors are still necessary, entailing a huge part of the area of the AFE. In addition, the passband gain of this amplifier is defined by its open-loop gain, which can be affected by variations in process, voltage, and temperature (PVT). Nevertheless, these variations are not detrimental to the recording of μECoG signals [28].

Transistor-based resistors

Efforts have been made to increase the resistance value to reduce the area of the input capacitors used to set the HPF pole. Obtaining high-value resistors requires long resistors with multiple folds to keep the layout compact, resulting in a large integration area. In addition to this size constraint, long resistors



(a) HPF pole set with biasing resistors (R_b).

(b) HPF pole set with autobiasing resistor (R_{fb}).

Figure 2.5: AC-coupled open-loop amplifiers.

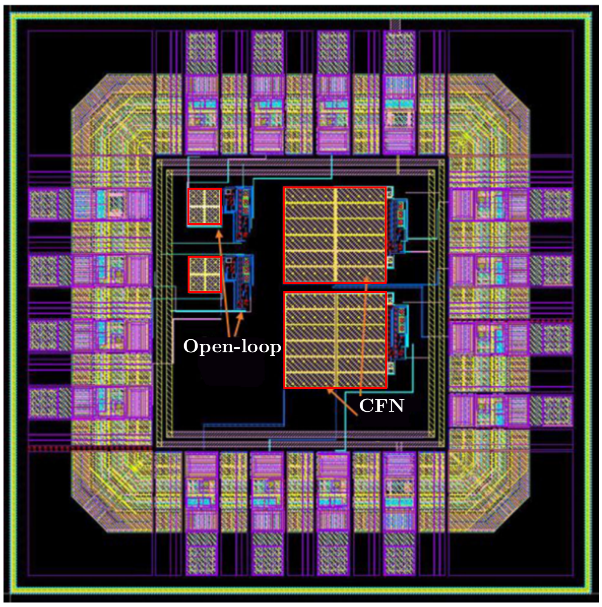


Figure 2.6: Layout of open-loop and CFN AC-Coupled amplifiers implemented in [67]. The input capacitors are highlighted in red.

introduce large parasitic capacitances that degrade their frequency response [68].

All these disadvantages led to the investigation of alternative solutions for the fabrication of high-value resistors. These solutions are based on circuit structures employing transistors, which exhibit a specific voltage-current (V-I) relationship with very high equivalent resistance while occupying considerably less space than a physical resistor of equal value. These devices are known as pseudo-resistors [42], [69]. Figure 2.7 illustrates typical implementations of pseudo-resistors.

Despite their extensive usage in the literature [53], [55], [70]–[74], pseudo-resistors exhibit significant non-linearity and are highly sensitive to process, voltage, and temperature (PVT) variations. These variations can cause their resistance to fluctuate by a factor of 100 [75]. Such inconsistencies make pseudo-resistors unreliable when used in HPF, as they can result in an unpredictable pole, leading to the potential loss of low-bandwidth neural signals.

An alternative to this approach is used in [51], [75], [76], where duty-cycled resistors are chosen instead

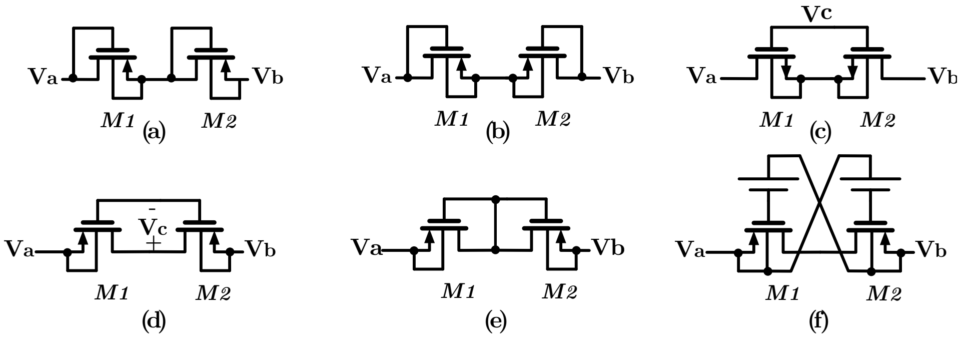


Figure 2.7: Pseudo-resistor architectures disclosed in the literature. (a) Two series connected MOS non-tunable pseudo-resistors [53]. (b) Two series outwardly connected gates MOS non-tunable pseudo-resistor [70]. (c) Symmetrical NMOS with source and gate in common [69]. (d) Two series connected MOS gate-voltage controlled pseudo-resistor [71]. (e) Two series inwardly connected gates MOS non-tunable pseudo-resistor [72]. (f) Balanced tunable pseudo resistor [55].

to set the pole. A duty-cycled resistor consists of a passive resistor placed in series with a switch (shown in Figure 2.8).

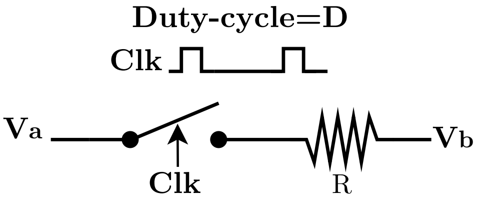


Figure 2.8: Circuit diagram of duty-cycled resistor.

When the switch is driven by a clock with a duty cycle of D, the equivalent resistance is then given by $\frac{R}{D}$. This results in large linear resistors in a small chip area. In [75], an equivalent resistance of 40 $G\Omega$ was obtained by employing a passive resistor of 1 $M\Omega$ and a D of 1/40000 with a clock frequency of 25 kHz. Although duty-cycle resistors are a good alternative to pseudo-resistors, there is still a limitation on their value due to leakage and noise constraints. Therefore, this approach still requires a considerably big input capacitor (20 pF in [75]) to set the HPF below 0.5 Hz.

2.1.2. DC coupling

To reduce the area overhead of the input capacitors used to set the HPF in AC-Coupled architectures (subsection 2.1.1), DC-coupled approaches have also been implemented in the literature. In such approaches, the LNA amplifies in an open-loop configuration. To block the EDO, a low-pass filter (LPF) is used in the feedback (which tracks the offset) and subtracts from the input to provide a high-pass filter as the overall transfer function [39]. This EDO rejection approach is known as DC servo loop (DSL) and is illustrated in Figure 2.9. Two types of DSL implementations can be distinguished regarding how the LPF is applied.

Analog feedback

The low-pass filter (LPF) implemented in the DSL can be realized by using either passive RC networks (same principle shown in subsection 2.1.1) or active integrators. Active integrators achieve large time constants with a reduced capacitor footprint, leveraging the Miller effect (shown in Figure 2.10) [77]–

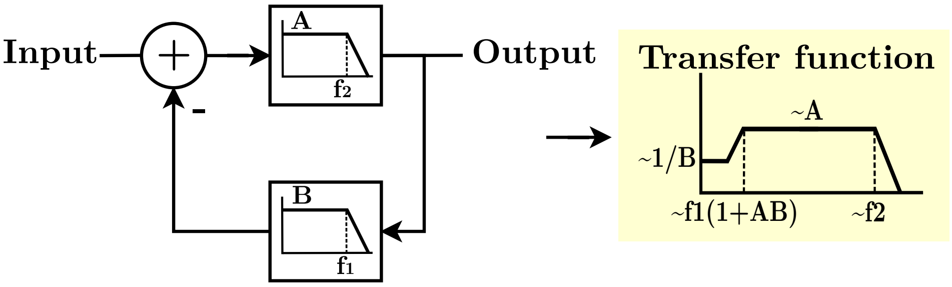


Figure 2.9: System representation of a DSL.

[79].

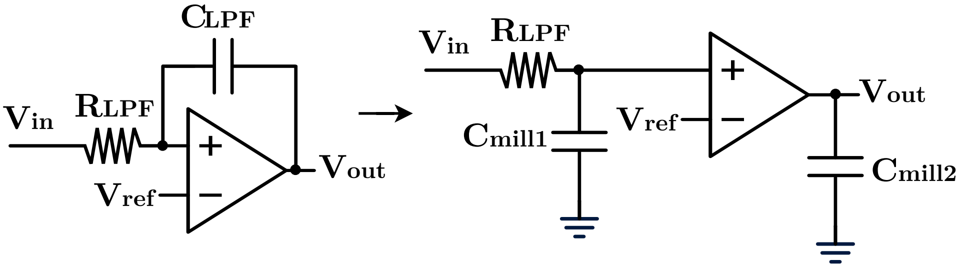


Figure 2.10: Representation of the miller effect in active integrators used in literature to set the LPF pole of the DSL.

The equivalent input miller capacitor C_{mill1} has an increased capacitive value equal to $C_{LPF} \cdot (1 + A)$, where A is the open loop gain of the amplifier used in the integrator topology. The LPF pole is set with the equivalent miler capacitor C_{mill1} and R_{LPF}. However, these implementations still rely on using transistor-based resistors (Figure 2.1.1), resulting in inaccurate LPF poles. Additionally, the integrator introduces noise [77] and increases the area and power consumption of the AFE (additional 0.01 mm^2 and 2.81 μW respectively, in [80]).

Mixed-signal feedback

A mixed-signal implementation does not require large capacitors or transistor-based resistors, as the LPF can be implemented in the digital domain. A digital LPF provides easier programming to adjust the pole and enables a well-defined high-pass pole frequency. Additionally, the digitally programmed poles are robust to PVT variations and result in a smaller chip area [81], [82]. Figure 2.11 represents the implementation of a mixed-signal feedback DSL.

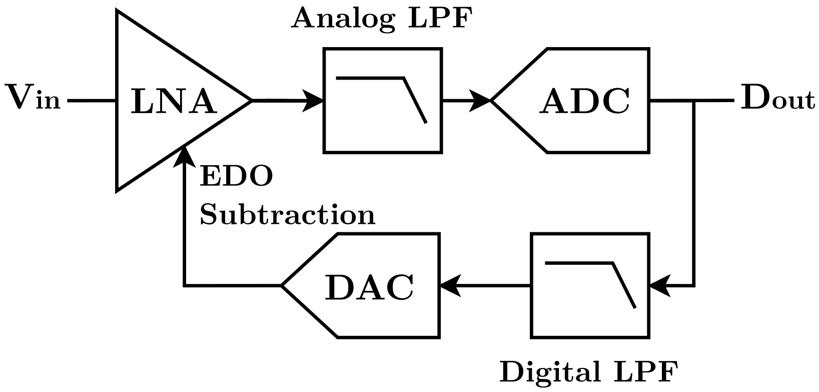


Figure 2.11: System representation of a mixed-signal DSL.

The ADC's digital output is processed by a digital LPF which tracks the EDO and subtracts it from the input of the signal chain employing a digital-to-analog converter (DAC). The EDO subtraction has been implemented in multiple ways in the literature.

Using current subtraction techniques is proposed in [81], [83], [84]. In [81] a 8-channel neural recording AFE is implemented. This design uses a mixed-signal DSL for each one of the channels containing a current-steering DAC (IDAC). Figure 2.12 illustrates the principle of the current subtraction operation.

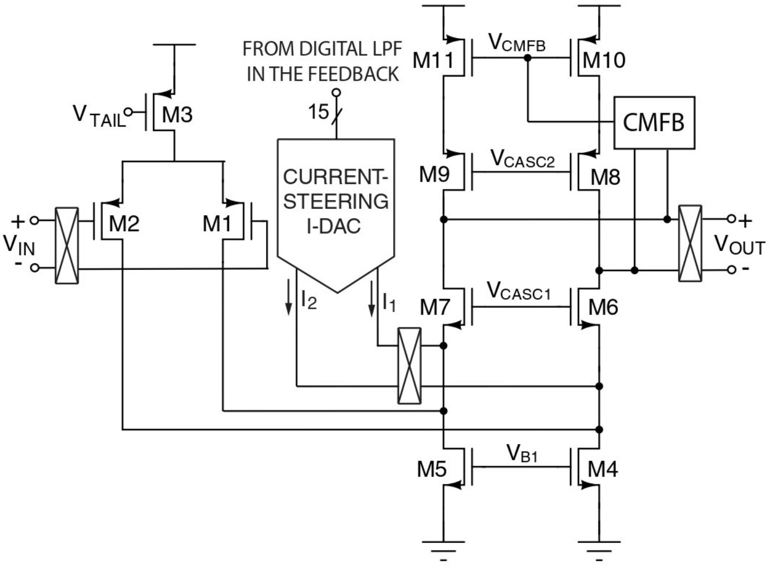


Figure 2.12: Subtraction of the EDO by using an IDAC in the folding nodes of the OTA [81].

In this design, the DSL injects current proportional to the EDO into the amplifier's internal nodes. This technique effectively cancels the EDO, leading to a compact design with an area of just 0.018 mm^2 per channel. However, while eliminating the need for input capacitors and reducing area, this approach introduces significant thermal noise (as the DAC noise is directly input-referred) and has a considerably high power consumption of 10 mW.

Another technique for EDO cancellation involves asymmetric tuning of the internal nodes within the input amplifier. This approach generates an input-referred offset (IRO) that counterbalances the EDO. In [85], [86], the degeneration resistor (R_s) of the amplifier was adjusted to create the IRO (Fig. 2.13). This method uses the DAC code to select the position of the current source within a resistor array, effectively embedding an offset into the transistor M_{1P}. However, this technique suffers from a large required area for the resistor array, occupying 44% of the total area per channel.

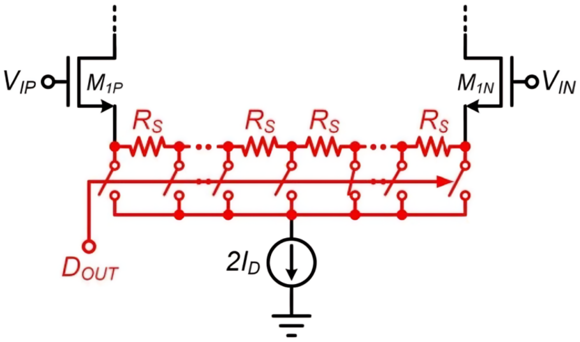


Figure 2.13: Cancellation of the EDO by controlling the position of the R_s array [86].

To avoid introducing additional components that increase noise or area, in [82], the EDO cancellation DAC is embedded directly into the differential pair of the amplifier. In this design, the input transistors are implemented with programmable widths (Fig. 2.14).

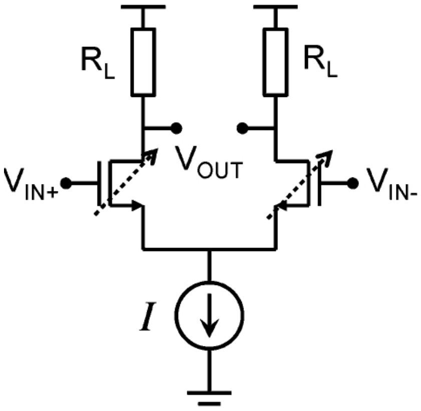


Figure 2.14: Cancellation of the EDO by asymmetrically controlling the width of the input transistors [82].

This approach employs a split input pair, with each side consisting of parallel transistors encoded with increasing unit size. A feedback loop dynamically adjusts the relative size of the two input pair transistors. It achieves this, by connecting or disconnecting parallel transistors from each side until the offset introduced by this asymmetry cancels the EDO offset (Figure 2.15 illustrates an example with all the parallel transistors sized equally and where the EDO is canceled by a width ratio of $W_{M1} = 1/4W_{M2}$).

The amplifier with the embedded DAC obtained an area of 0.0037 mm^2 and a power of 4.13 μW without introducing any additional noise. However, this approach is not compatible with flicker noise cancellation techniques.

Lastly, another approach is known as body-induced offset cancellation, which leverages the body contact

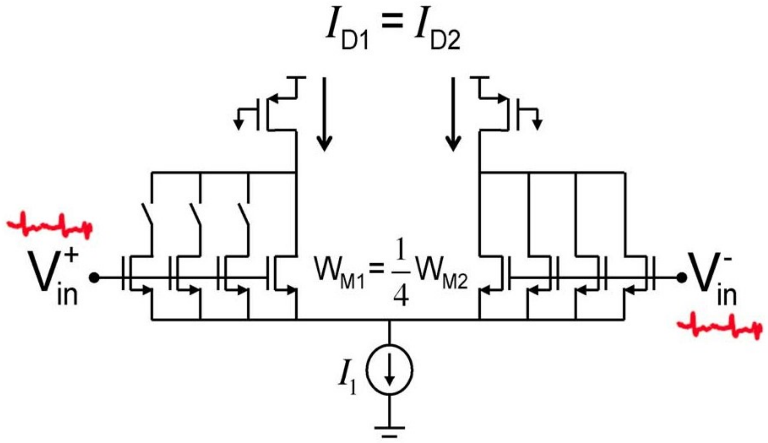


Figure 2.15: Cancellation of the EDO by enforcing a width ratio of 1/4 for the input transistors) [82].

of the input transistors [29], [80], [87]. With this implementation, the DSL modulates the bulk voltage, generating an IRO that counteracts the DC offset. The generated IRO offset in this case can be derived as shown in Equation 2.4.

$$IRO = \frac{g_{mb}}{g_m} \cdot V_{dsl} \tag{2.4}$$

Where $\frac{g_{mb}}{g_m}$ is the ratio between the bulk and gate transconductances of the input transistors (approximately 1/4 in [80]), and V_{dsl} is the output swing of the DSL. This ratio suggests that V_{dsl} must be around 4 times larger than the targeted EDO compensation range. Additionally, it also poses an advantage as the noise introduced by the DSL is attenuated to 1/4 when referred to the input of the amplifier. When employing body-induced feedback, care needs to be taken to avoid forward biasing the body diode of the input transistors, which limits the operation range of the DSL, consequently limiting the range of compensation for the EDOs.

While this body-controlled feedback technique is used in architectures like [80], [87], these designs still rely on setting the low-pass filter (LPF) using large time constants, as shown in Figure 2.16. In [80], an area of 0.02 mm^2 per channel was achieved, with the DSL circuit itself consuming half of that area.

An alternative approach is presented in [29]. This implementation employs a mixed-signal EDO compensation while embedding the DAC into the amplifier (Fig. 2.17).

While body biasing in regular CMOS technology is sensitive to the possible forward biasing of the bulk diodes, fully depleted silicon on insulator (FDSOI) technology allows for a wide range of body biasing (from -2 V to 2 V, as noted in [29]). This wide range is leveraged by directly feeding the digital code to the bulk terminals. This approach splits the input transistors into binary-coded parallel devices, whose bulk connections are tied to their respective codes. This method does not introduce noise or increase the area, as no additional components are required. However, an equivalent implementation compatible with standard CMOS technology has not been developed to date.

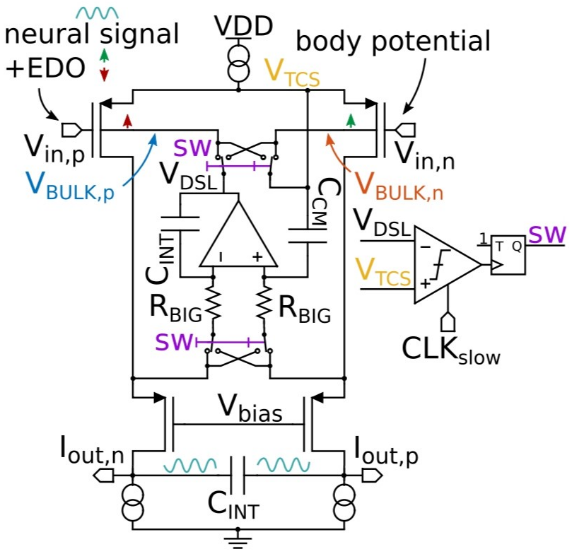


Figure 2.16: subtraction of the EDO using a buk modulation approach and an analog integrator to set the LPF pole [80].

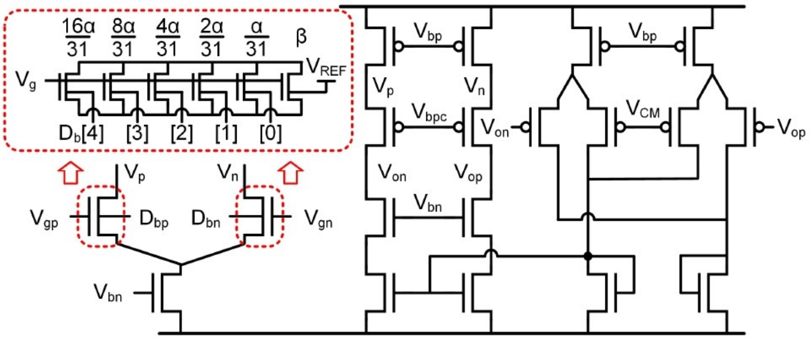


Figure 2.17: Subtraction of the EDO by a buk modulation approach where the digital codes of a 5-bit accumulator are directly applied to the body connections [29].

2.2. Noise

As highlighted in chapter 1, the neural signals have a very low amplitude, requiring a low input-referred noise (IRN) to differentiate the signal of interest. For this reason, the LNA has to be carefully designed to amplify the signals while introducing minimal noise.

While the noise contribution of the LNA highly depends on the chosen topology, the input transistor pair typically entails the maximum noise contribution [38]. CMOS transistors exhibit thermal noise due to the random motion of charge carriers (electrons and holes). This random motion results in fluctuations in the current flowing through their conductive channel, which manifests as thermal noise [88]. The power spectral density (PSD) of the thermal noise voltage at the gate of a CMOS transistor is expressed as shown in Equation 2.5.

$$\overline{v_n^2} = \frac{k_B T \gamma}{g_m} \tag{2.5}$$

Where k_B is the Boltzmann constant (1.38×10^{-23} J/K), T is the absolute temperature in Kelvin, γ is a factor that depends on the operation region of the transistor (typically around 2/3 for long-channel CMOS in saturation, but it can be higher for short-channel devices) [88], and g_m is the transconductance of the CMOS device.

Additionally, CMOS transistors also suffer from flicker noise (also known as 1/f noise). This noise arises due to the random trapping and release of the charge carriers by the imperfections in the semiconductor material. These random events cause fluctuations in the current, particularly noticeable at lower frequencies [88]. The PSD of the flicker in a CMOS transistor is exposed in Equation 2.6

$$\overline{v_n^2} = \frac{K_f}{C_{ox} \cdot WL} \cdot \frac{1}{f} \tag{2.6}$$

Where K_f is the flicker noise coefficient (process-dependent), C_{ox} is the gate oxide capacitance per unit area, W and L are the width and length of the CMOS device and f is the frequency. Flicker noise becomes the dominant noise factor in low-frequency neural recording systems because the noise level is inversely proportional to the frequency [39]. Several techniques have been employed to reduce the thermal noise floor and attenuate the problematic 1/f noise.

2.2.1. Circuit techniques

To minimize the thermal noise contribution of the LNA, the g_m of the input transistors is maximized. To achieve this, the input pair is biased in the sub-threshold region by increasing the W/L ratio for a constant biasing current [38], [88]. Additionally, current reuse techniques can be employed to further reduce thermal noise. By recycling the current through multiple stages of amplification, it is possible to enhance the overall transconductance without a proportional increase in power consumption, thereby achieving lower thermal noise [89].

On the other hand, to reduce the flicker noise contribution, the differential pair transistors are chosen to be PMOS type (which have lower K_f) and according to Equation 2.6, the size (W and L) is increased, resulting into a considerable area overhead [90].

2.2.2. Chopper stabilization

To reduce the effects of the flicker noise without the above-mentioned area penalty, chopper stabilization (CHS) is usually used. Figure 2.18 illustrates its principle.

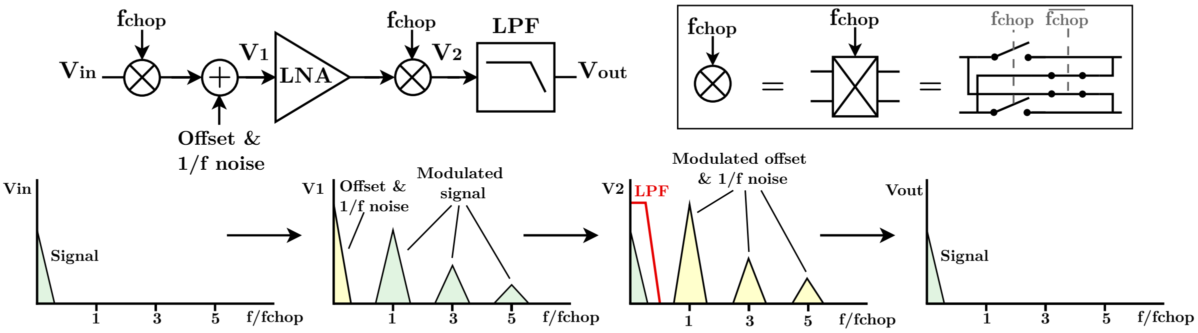


Figure 2.18: Operation principle of the chopping stabilization technique.

The first chopper (implemented by switches) modulates the input signal to a higher frequency (f_{chop}) where the 1/f noise is not present. In the next step, the amplifier amplifies the input signal (while adding a DC offset and 1/f noise, present in V_1). After that, the output chopper modulates both the

DC offset and the 1/f noise, while the input signal is brought back to the baseband (this signal is present at V_2). The up-modulated offset and 1/f noise appear as output ripples, which must be eliminated, a straightforward way of doing this is by employing a LPF at the output of the amplifier.

While this approach effectively eliminates the flicker noise, it also decreases the input impedance of the recording AFE. The input chopper switches, along with the LNA input capacitance, can be modeled as an equivalent resistor [38], thereby highly reducing its input impedance if a high f_{chop} is employed. The equivalent input resistance of the AFE when using chopping stabilization is shown in Equation 2.7.

$$Z_{in} = \frac{1}{f_{chop} \cdot C_{eq,in}} \tag{2.7}$$

where $C_{eq,in}$ is the equivalent capacitor formed by the input capacitor (in the case of AC coupling) and the parasitic capacitances at the input of the LNA. Using the CHS technique for an AC-coupled architecture can result in an unacceptably small input impedance (8 $M\Omega$ in [91]). Therefore, CHS is better suited for DC-coupled architectures where C_{eq} is only the parasitic input capacitance of the LNA.

To increase the input impedance of the AFE, additional positive feedback is implemented in [75], [92]–[94] (Fig. 2.19). However, this approach introduces additional area, noise, and power consumption.

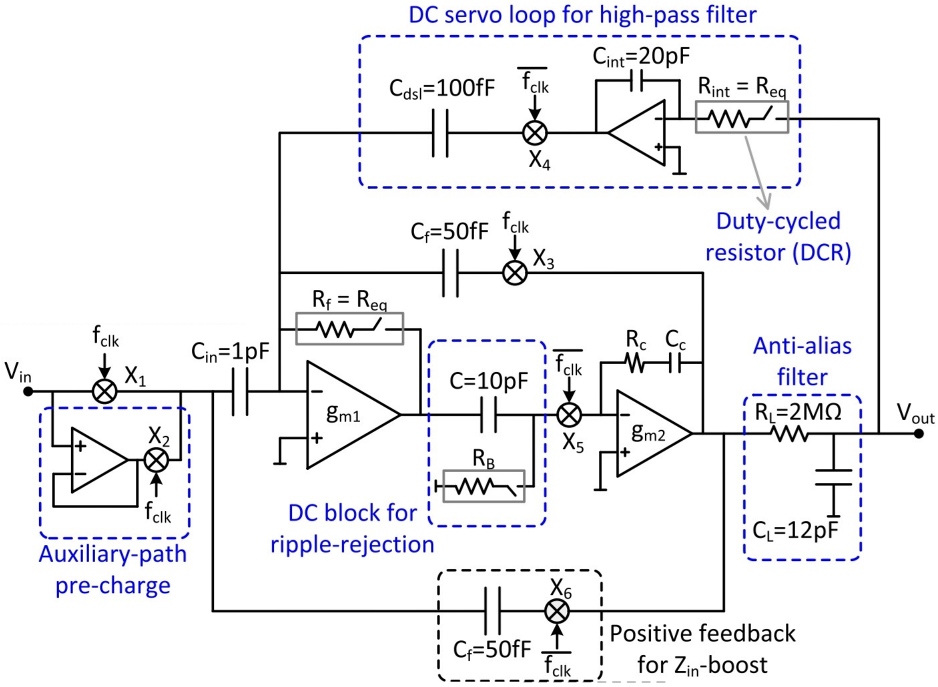


Figure 2.19: Chopping implementation with input capacitors. The circuit uses an additional positive feedback loop and DSL [75].

Additionally, in this topology, the input choppers up-modulate the EDO, which is no longer filtered by the HPF pole set with the input capacitor. To eliminate the up-modulated EDO in AC-coupled AFEs, an additional DSL is implemented. An analog DSL is used in [75], [92]–[94], (Fig. 2.19). On the other hand, a mixed-signal DSL can also be implemented instead, by tracking the EDO with a digital LPF and then subtracting it by capacitive DAC (CDAC) using the input capacitors as summing nodes. Although the input capacitors do not have to be sized according to the required time constant of the circuit, they still have a minimum size requirement as their capacitance needs to be bigger than the surrounding parasitic capacitances (which leads to an increased circuit area) [29], [34].

2.3. Anti-aliasing filter and sampling

Following the EDO rejection via AC coupling or DSL approaches and the signal amplification performed by the Low Noise Amplifier (LNA), the signal must be low-pass filtered (see Figure 2.1). This step limits the bandwidth to avoid signal aliasing and attenuates all unwanted higher-frequency components. Subsequently, the signal is sampled before being quantized by the ADC.

2.3.1. Anti-aliasing low pass filter

Due to the low-frequency spectrum of the neural signals (exposed in chapter 1), the design of passive or active RC filters requires large time constants [95]. As previously mentioned in subsection 2.1.1, these large time constants often require huge passive elements or set unreliable pole locations. By contrast, switched-capacitor filters (SCF) exploit the switched-capacitor resistor concept [95]. This approach is shown Figure 2.20.

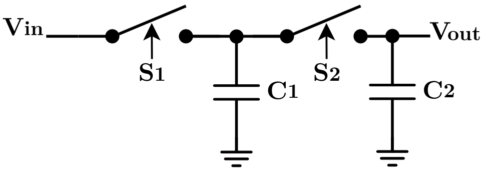


Figure 2.20: SC-LPF circuit.

The equivalent resistance R_{eq} of the switched capacitor is derived as:

$$R_{\mathrm{eq}} = \frac{V_{\mathrm{in}} - V_{\mathrm{out}}}{I} = \frac{1}{f \cdot C_1} \tag{2.8}$$

The $f_{-3dB,SC-LPF}$ is then:

$$f_{-3dB,SC-LPF} = \frac{1}{2\pi R_{eq} C_2} \tag{2.9}$$

SC-LPFs are highly useful for the recording of low-frequency signals (such as neural signals) because $R_{eq} \propto \frac{1}{C_1}$. Therefore, the area required for this resistor decreases as R_{eq} increases [95]. Additionally, the pole frequency is only determined by the capacitor ratios, which can be precisely controlled.

2.3.2. Sampling

Sampling is essential before feeding an analog signal to an ADC (where digital quantization takes place). This process, typically performed by a sample and hold circuit (shown in Figure 2.21), converts the continuous signal into discrete voltage readings suitable for ADC. The sampling process introduces thermal noise, known as KT/C noise, which is inversely proportional to the sampling capacitor's capacitance. This noise originates from the on-resistance of the sampling switch (R_{on}) and the sampling capacitor (C_{samp}). The derivation of the noise power at the output of the sampling circuit provides insight into its origin:

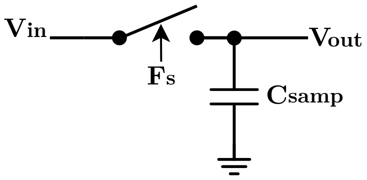


Figure 2.21: Sampling and hold circuit.

$$\overline{v_{\text{out,tot}}^2} = \int_0^{\infty} 4kTR_{\text{on}} \left| \frac{1}{1 + j2\pi f R_{\text{on}} C_{\text{samp}}} \right|^2 df = 4kTR_{\text{on}} \cdot \frac{1}{4R_{\text{on}} C_{\text{samp}}} = \frac{kT}{C_{\text{samp}}} \tag{2.10}$$

where k is Boltzmann's constant and T is temperature. Moreover, noise folding is a critical consideration in the sampling process. This phenomenon occurs when high-frequency noise components are aliased into the Nyquist band during sampling. The effect extends to all frequencies up to the bandwidth of the sampler, which is typically much higher than the Nyquist frequency.

2.3.3. Windowed Integrated sampling

Windowed integrated sampling (WIS) or Boxcar sampling, combines the amplification phase with the antialiasing and sampling functions in a single block (Fig. 2.22), which can save considerable power and silicon area [95]. In this implementation, the input signal is converted to an output current (I_{out}) by the OTA. I_{out} is integrated for a fixed time window (T_{int}) in a capacitor (C_{int}) and the resulting integrated voltage is taken as a sample [96]. After that, C_{int} is reset by the RST switch to clear the charge before the next sampling phase [97]. Such sampling process essentially operates in the charge domain, while the output signal is in the voltage domain. Its transfer function is exposed in Equation 2.11.

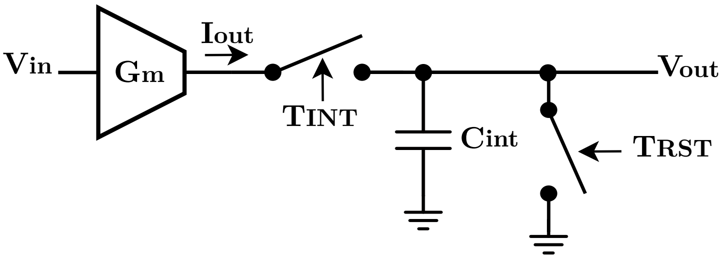


Figure 2.22: Circuit representation of a boxcar sampler.

$$V_{\text{out}}\ (nT_{\text{s}}) = \frac{g_m}{C_{int}} \int_{nT_s}^{nT_s + T_{int}} V_{in}(t)dt \tag{2.11}$$

Where the g_m is the trans-conductance of the OTA, and T_s is the sampling period. The transfer function of a boxcar sampler can be rewritten as shown in Equation 2.12 [97].

$$H(f) = \frac{g_m T_{int}}{C_{int}} \cdot \frac{1 - e^{-j2\pi f T_{int}}}{j2\pi f T_{int}} \tag{2.12}$$

which can be interpreted as a convolution integral of V_{in} and a rectangular window with a height of $\frac{g_m}{C_{int}}$ and a width of T_{int}. The ideal magnitude transfer function (assuming infinite OTA output impedance) can be expressed as a sinc-type LPF (unlike the first-order LPF of conventional sampling circuits) with a defined DC gain (eq. 2.13), provided that T_{int} occupies most of T_s with only a brief reset phase.

$$|H(f)| = \frac{g_m T_{int}}{C_{int}} \left| \frac{\sin(\pi T_s f)}{\pi T_s f} \right|. \tag{2.13}$$

Equation 2.13 defines a LPF with nulls at integer multiples of the sampling frequency, a main lobe at dc with a dc gain of $\frac{g_m T_{int}}{C_{int}}$ and a set of side-lobes rolling off at -20 dB/dec (Fig. 2.23).

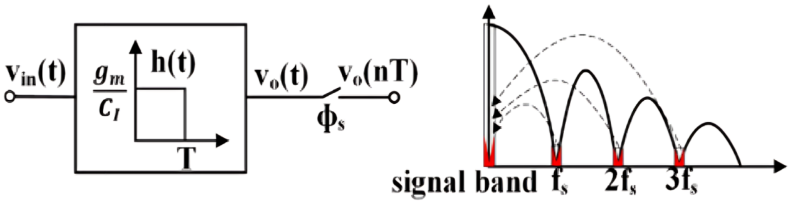


Figure 2.23: Equivent transfer function and frequency response of the boxcar sampler.

Implementing a boxcar sampler provides the desired anti-aliasing filtering without introducing additional large passive elements to set the LPF pole [98].

It's important to note that the ideal transfer function, which assumes an infinite output impedance (r_o) of the OTA, would result in perfect nulls (reaching negative infinity) completely eliminating noise folding. However, in practice, the finite r_o of the OTA causes these nulls in the frequency response to be finite, not reaching negative infinity. As a result, some degree of noise folding occurs, although it is significantly reduced compared to conventional sampling techniques. In addition, the *RST* switch introduces the previously mentioned kT/C noise. However, this noise is highly attenuated by the gain of the boxcar sampler when referred to the input.

Moreover, variations of the g_m only affect the gain of the boxcar, but not the frequency response [99]. For these reasons, the boxcar sampler has been chosen in multiple neural recording designs in literature [29], [34], [51], [82].

2.4. Analog to Digital Conversion (ADC)

Following the amplification of neural signals and attenuation of the EDO (as detailed in section 2.1), their bandwidth is limited and sampled (explained in section 2.3). This allows for their conversion into digital form by the Analog-to-Digital Converter (ADC) (Fig. 2.1). Multiple topologies of ADCs are found in the literature and are explained below.

2.4.1. Oversampled ADC

Oversampled ADCs sample input signals above the Nyquist rate, spreading the quantization noise which can then be filtered to increase the ADC's overall SNR [100]. Assuming the quantization noise of an N-bit ADC is approximated as white noise, the increase in SNR is derived as in Equation 2.14.

$$\Delta\text{SNR} = 10\log_{10}(\text{OSR}) \tag{2.14}$$

where OSR is the oversampling ratio ($f_s/f_{nyquist}$).

Sigma-delta modulators ($\Delta\sum$) are a common implementation of oversampling ADC (Figure 2.24) that also benefit from noise shaping. Noise shaping pushes quantization noise to higher frequencies outside the signal band, allowing for an improved SNR after filtering.

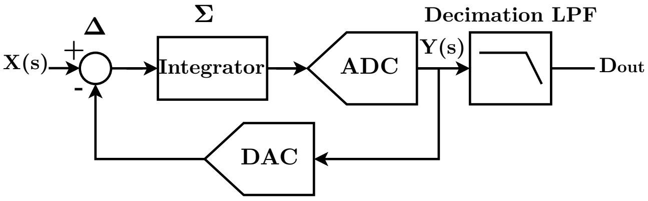


Figure 2.24: Basic architecture of a sigma-delta modulator.

For a nth-order sigma-delta ADCs, the SNR improvement is:

$$\Delta\mathrm{SNR} = 20\log_{10}\left(\frac{\mathrm{OSR}^{(\mathrm{n}+0.5)}\sqrt{2\mathrm{n}+1}}{\pi^{\mathrm{n}}}\right) \tag{2.15}$$

In neural recording, oversampled ADCs are a popular approach, offering advantages like avoiding the need for a low-noise amplification stage [39], [101]–[111]. These designs are often referred to as direct-to-digital converters (DDC) and leverage Time Division Multiplexing (TDM) to integrate a high channel count per ADC, reducing power consumption and area overhead. However, recording neural signals superimposed on EDOs requires a large dynamic range, necessitating higher-order modulators and increased OSR, leading to greater power consumption. In [112], a $\Delta^2\sum$ structure adds an extra Δ stage to a $\Delta\sum$ modulator. As a result, the difference of two subsequent input signals is fed into the quantizer, eliminating the DC component but requiring additional area (5.98 mm^2/channel). Alternatively, EDOs can be eliminated by AC-coupling (as explained in subsection 2.1.1) [108], [110] or mixed-signal DSLs [29], [34]. Overall, DDC architectures can achieve extremely low area per channel (e.g., 0.001 mm^2/channel as reported in [29]). However, they heavily rely on TDM to share a single DDC across multiple channels, with 16 channels being shared in the case of [29].

2.4.2. SAR

Successive Approximation Register (SAR) ADCs have become increasingly popular for neural recording AFE due to their moderate resolution of 8-10 bits, a sampling speed ranging from 1 to 500 kS/s, and high power efficiency [39]. Figure 2.25 illustrates the basic architecture of a typical SAR ADC. The

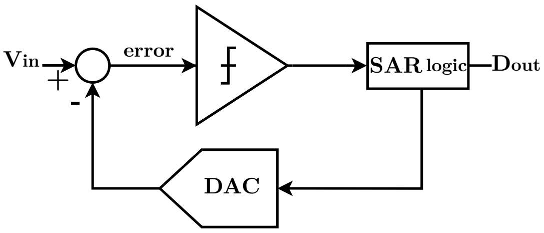


Figure 2.25: Basic architecture of a SAR ADC.

SAR DAC, which occupies a significant area on the chip, also leads to higher power consumption when switching. Various strategies have been suggested to minimize the DAC's chip area. For example, the use of a bridge capacitor with a compact design [113] and the implementation of a unit-length DAC [114]. Additionally, numerous efforts have focused on reducing the DAC's switching power [115]–[117]. Despite these advancements, most SAR ADC designs in the literature employ the ADC across different channels using TDM techniques [118], [119].

2.4.3. Single-slope ADC

Single-slope (SS) ADC, also known as Ramp ADC is the simplest possible implementation for an ADC. The input signal (V_{in}) is compared to an increasing voltage ramp (V_{ramp}), while a counter measures the number of clock cycles until V_{ramp} matches V_{in}. This method involves a ramp generator, comparator, and counter. However, it requires a precise and stable ramp generator to maintain ADC linearity across PVT (process, voltage, and temperature) variations. Achieving high resolution demands a high-frequency clock since the resolution is determined by 2^N clock periods per conversion. Consequently, SS-ADCs are mainly used for 6-10 bit resolutions. This ADC is an appropriate solution for low-speed and low-power applications. Figure 2.26 shows the basic implementation of this ADC. This approach has been effectively applied in neural signal recording, as detailed in [51]. In this design, 1024 channels are employed to record single-cell neural activity, sharing a globally distributed differential ramp. The result is impressively low area and power usage (0.00129 mm^2 per channel and 0.268 μW per channel, respectively), all achieved without the need for TDM techniques.

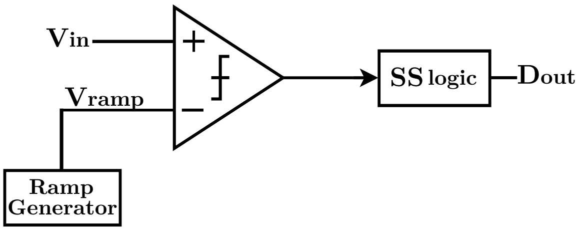


Figure 2.26: Basic architecture of a SS ADC.

2.5. Comparison of neural recording AFEs

To provide a thorough comparison of existing analog front-end topologies that favor an orthogonal design, each key element conforming to an AFE is evaluated independently. In the tables below (Tab. 2.1, 2.2 and 2.3), a '-' sign indicates a negative effect on the row parameter, whereas '+' indicates a positive effect.

1. **EDO compensation and LNA:** Table 2.1 summarises the configurations employed to attenuate the EDO discussed in the previous section, specifically in terms of area, gain accuracy, reliability of the HPF pole, power consumption, and input impedance.

Table 2.1: Comparison of different EDO compensation and LNA configurations.

Parameter	**AC-Coupling**		**DC coupling**	
	Closed loop	**Open-loop**	**Analog FB**	**Mixed Signal FB**
Area	--	+	+	++
Gain stability	++	-	-	-
HPF reliability	-	-	-	++
Power	++	++	--	+
Zin	--	-	++	++
CHS compatibility	--	-	++	++

While AC coupling (employed in both closed-loop and open-loop configurations) and DC coupling circuits with analog feedback loops are common approaches, they face limitations in high-density multi-channel applications. These limitations stem from the need for large time constants. Mixed-signal DSL circuits offer a promising alternative for dense recording arrays. This approach avoids the use of large time constants and set reliable HPF poles, making it a highly desirable alternative. Table 2.2 compares the previously exposed mixed-signal DSL configurations.

Table 2.2: Comparison of different configurations of mixed-signal DSL.

Parameter	**IDAC** [81], [83], [84]	**RDAC** [85], [86]	**WMOD** [82]	**BMOD** [29]
Area	-	-	++	++
Noise	-	-	++	++
CHS compatibility	Yes	Yes	No	Yes
STD CMOS compatibility	Yes	Yes	Yes	No

Width modulation and body modulation, as demonstrated in [29], [82] are particularly attractive solutions because they do not introduce extra noise, power, or area in its implementation. However, width modulation is not compatible with the chopper stabilization technique which is crucial for low-area μECoG recording. On the other hand, while the bulk modulation implemented in [29] does not present this problem, it has only been implemented in FDSOI technology. Therefore, efforts should be made to adapt this concept to be compatible with standard CMOS technology.

2. **Noise reduction techniques** Noise reduction circuit techniques must be considered when designing the AFE to lower the thermal noise of the system. In addition, chopper stabilization is a crucial approach to reduce the flicker noise of the LNA without increasing the area requirement of the input transistors.
3. **Amplification, sampling and anti-aliasing:**

   Window-integrated sampling is the optimal approach for amplification and sampling due to its compactness, built-in anti-aliasing filtering, and minimal noise folding during sampling. It is also inherently compatible with DC-servo loop architectures that use open-loop OTA configurations.
4. **Analog to Digital conversion:** Table 2.3 presents the comparison between the ADC architectures discussed in the section.

Table 2.3: Comparison of different ADC topologies.

Parameter	**Oversampled**	**SAR**	**Single-Slope**
Resolution	++	+	+
Conversion speed	+	++	-
Area (Single CH/AFE)	--	-	++
Power (Single CH/AFE)	--	-	+
Design complexity	--	-	++

Oversampled ADCs, such as sigma-delta modulators, are renowned for their impressive performance. However, they heavily rely on multiplexing techniques, which are not well-suited to a distributed single-channel AFE approach. In high-channel-count applications, implementing a dedicated modulator for each channel can lead to an increased area overhead. By contrast, single-slope ADCs utilizing a shared ramp across all recording channels offer a power and area-efficient alternative that is compatible with an array of single-channel AFEs.

Considering the above points, the AFE designed in this thesis is a DC-Coupled chopper-stabilized low-noise boxcar sampler (LNB) with a mixed-signal body modulation EDO compensation loop and an SS ADC. The system architecture of the design is explained in chapter 3 and the circuit implementation can be found in chapter 4.

3

System architecture

In this chapter the system level design of the proposed AFE for the recording of μECoG signals is presented and discussed (Fig. 3.1).

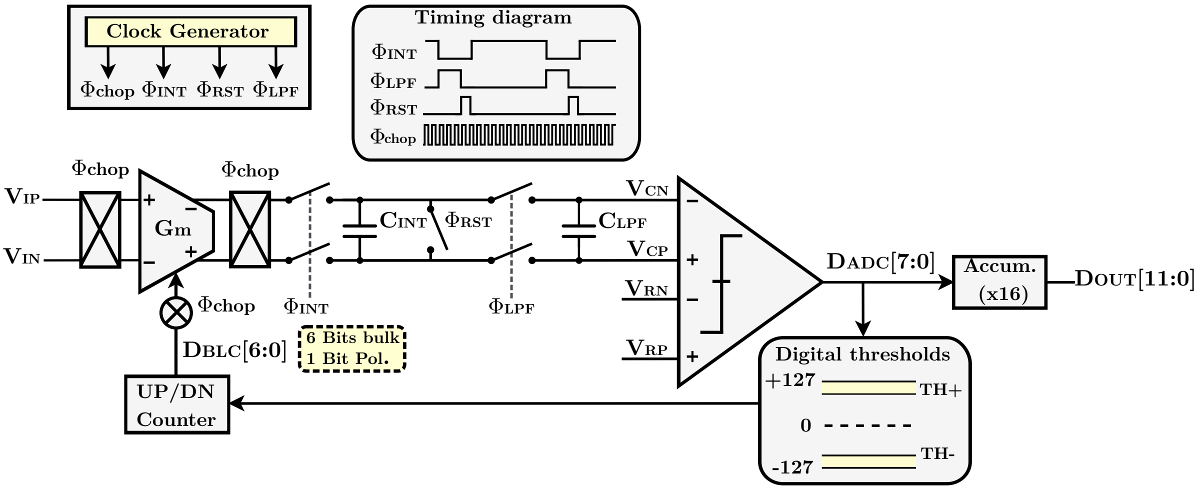


Figure 3.1: Proposed neural recording system architecture.

The proposed AFE features a DC-coupled chopper-stabilized low-noise boxcar sampler (LNB) followed by a passive switched-cap low-pass filter (SC-LPF) and a single-slope (SS) analog-to-digital converter (ADC). The positive input connection of the AFE (V_{IP}) is connected to the corresponding recording electrode, while the negative input connection (V_{IN}) is connected to the shared reference electrode.

The LNB minimizes noise folding and improves anti-aliasing by introducing notches at multiples of the sampling frequency, attenuating the chopping ripple ($f_{chop} = 3f_s$). Further high-frequency noise reduction is achieved by reusing the LNB integration capacitor ($\mathrm{C_{INT}}$) as a switched-capacitor equivalent resistor to form a passive low-pass filter with the sampling capacitor ($\mathrm{C_{LPF}}$).

A comparator compares the output of the SC-LPF with a globally distributed ramp to perform 8-bit quantization. The SS-ADC is oversampled (OSR = 16) to achieve a Nyquist-rate ($\mathrm{f_s}$ = 1 kSps) resolution of 12 bits. The ramp generator can be shared across all pixels in an array, significantly reducing the power consumption and chip area, as demonstrated in [51]. A novel EDO cancellation loop is employed to prevent the saturation of the LNB due to the large EDO, and its functionality is described in the next section.

3.1. Bi-level EDO compensation

The Bi-level compensation (BLC) of the EDO is a crucial part of the presented design, as it is the element that prevents the amplifier and consequently, the AFE from saturation, allowing a correct signal acquisition. The proposed Bi-level EDO compensation is comprised of two main parts.

3.1.1. Control loop

The proposed approach eliminates the need for the analog or digital low-pass filter typically used in DSL architectures to extract the EDO (explained in subsection 2.1.2). Instead, it compares the digital output of the ADC (D_{ADC}) with predefined upper and lower thresholds (set close to the full-scale range) to control the EDO cancellation loop, as described in [29] and [120].

In cases where the AFE input range exceeds the expected neural signal, a digital output near either extreme indicates a significant EDO at the input. Practically, if the output surpasses the upper threshold (or falls below the lower threshold), a digital accumulator (D_{BLC}) is incremented (or decremented) by 1 LSB (Fig. 3.2). The D_{BLC} directly adjusts the cancellation DAC to reduce the EDO within the AFE's linear range. The control loop also keeps track of the polarity of the EDO with an additional polarity bit (1 for a positive EDO and 0 for a negative EDO).

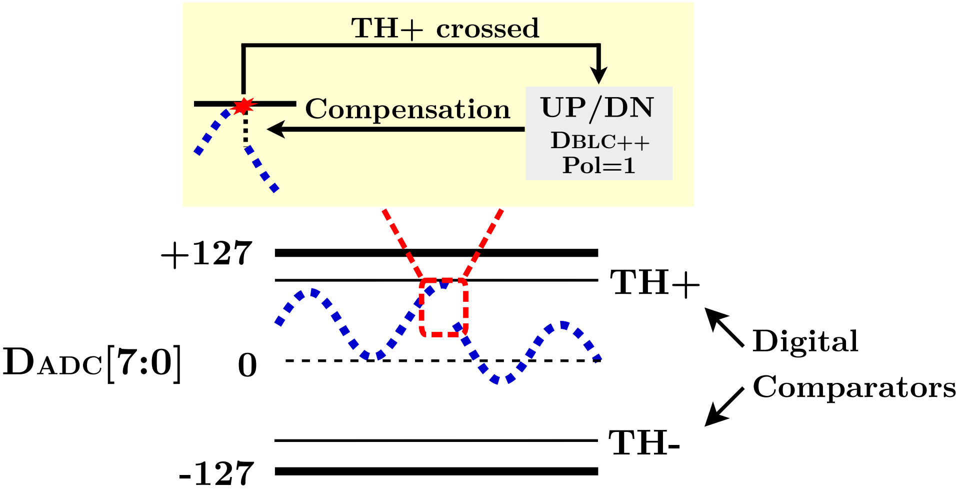


Figure 3.2: Representation of Bi-level compensation with a sinusoidal input signal and a gradually increasing EDO, crossing the upper threshold.

A code change in the EDO cancellation DAC introduces a DC jump (discontinuity) in the output signal. However, due to the slow and gradual nature of EDO drift, these discontinuities occur infrequently and can be easily filtered out, thereby having minimal impact on the acquired signal [121]. Since this approach does not require precise tracking and elimination of the EDO, a coarse DAC is suitable for this architecture. Nevertheless, a residual EDO will remain in the signal acquisition path, requiring additional input swing and redundancy in the ADC resolution, as part of the output codes will capture the residual EDO.

During the initial operation of the AFE, the output code will be in the saturation zone if a large EDO is present at the input. As it crosses one of the output digital comparators, it increases D_{BLC} each cycle. After a few samples, the accumulator reaches its optimal value, which brings the AFE into its linear region. From now on, this code is fixed until one of the two comparators is triggered again.

3.1.2. Embedded EDO cancellation DAC

The body terminal of the input transistors in the OTA is modulated to induce an offset proportional to the D_{BLC} code. PMOS devices are used to enable independent control of the body terminal. The input transistors are split into N binary weighted parallel devices with independent N-wells. Depending on D_{BLC}, each body terminal is connected to the voltage supply (V_{DD}) or a low-impedance node with $V_{B,BLC} < V_{DD}$ (Fig. 3.3).

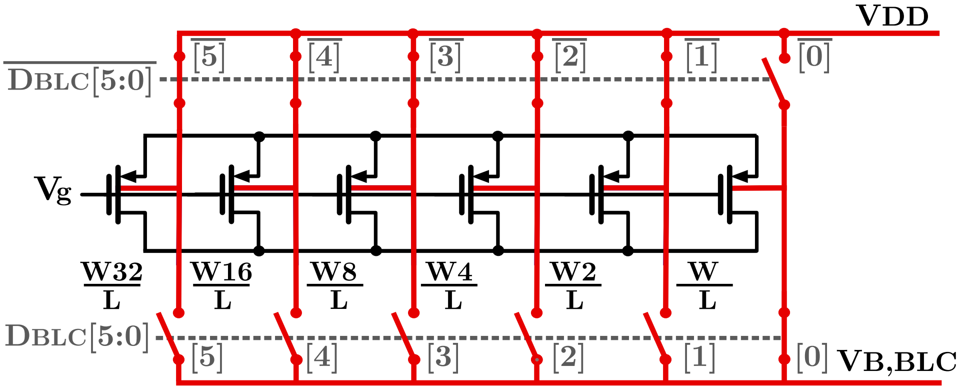

VDD
$\overline{[5]}$ $\overline{[4]}$ $\overline{[3]}$ $\overline{[2]}$ $\overline{[1]}$ $\overline{[0]}$
$\overline{\text{DBLC}[5:0]}$
Vg
W32/L
W16/L
W8/L
W4/L
W2/L
W/L
DBLC[5:0]
[5] [4] [3] [2] [1] [0]
VB,BLC

Figure 3.3: Example of the proposed bulk modulation for the input transistors with 6 bits and D_{BLC} = "000001".

A single-ended DAC approach is implemented for the bulk modulation. As a result, the body potential can only be adjusted in one direction (from V_{DD} to $V_{B,BLC}$), rather than around a common mode voltage. Consequently, the polarity of the compensation is encoded in the side where this modulation is implemented. To compensate for both positive and negative EDOs, the body terminals of the input transistors are controlled according to a polarity bit. For a positive EDO, the positive input transistors are controlled by D_{BLC}, while all body connections of the negative input transistors are connected to V_{DD} (equivalent to a D_{BLC} code of "000000"). The opposite configuration is used for a negative EDO. This arrangement enables bidirectional offset compensation despite the unidirectional nature of the body potential modulation inherent to the single-ended DAC approach.

The generated input-referred offset is derived as

$$IRO = \frac{g_{mb}}{g_m} \cdot \frac{\Delta_{V_b} \cdot D_{BLC,dec}}{2^{N_{BLC}}} \tag{3.1}$$

where g_{mb} is the backgate transconductance of the input transistor, $\Delta_{V_b} = V_{DD} - V_{B,BLC}$, $D_{BLC,dec}$ is the DAC control code in decimal base and N_{BLC} is the DAC number of bits. Δ_{V_b} must be carefully controlled to avoid forward biasing the body to drain/source junctions (Fig. 3.4).

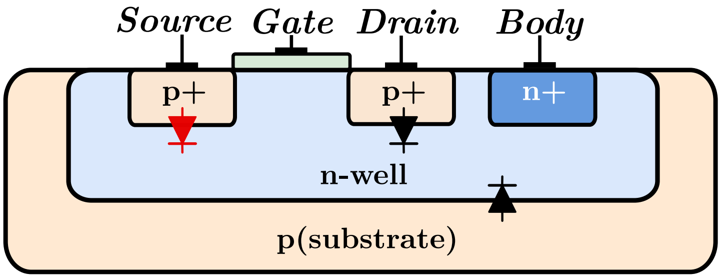


Figure 3.4: Cross-section of a PMOS transistor with pn-junction diodes, highlighting in red the diode most susceptible to forward biasing.

The diode between the substrate and body connection is always reverse-biased. However, the diodes between the source and bulk, and drain and bulk, can be forward biased. Since the source potential is usually higher than the drain potential in a PMOS device, the source-body diode is most susceptible to forward biasing and must be carefully controlled. Figure 3.5 illustrates the general I-V curve of the PMOS body-source diode and the approximated safe zone of operation.

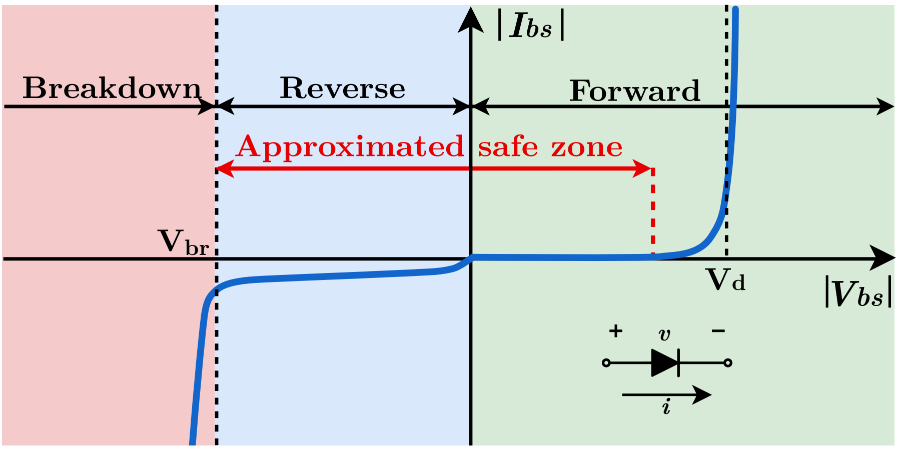


Figure 3.5: General representation of the I-V curve of the bulk-source diode of the PMOS input device.

Within the safe zone, a higher Δ_{V_b} allows for a higher EDO compensation range. At the same time, a lower Δ_{V_b} reduces the residual EDO after compensation. Splitting the input transistor into a larger number of devices can increase the compensation resolution, but also increase the overall area of the OTA.

Moreover, the chopper input switches up-modulate the input signal, which also up-modulates the EDO (see Fig. 3.1). Therefore, the cancellation DAC must also be chopped. This is achieved through a switching scheme where the bulk connections of the positive and negative input transistors (controlled by D_{BLC} or connected to V_{DD}) are alternated with the same frequency as their gates, ensuring that the bulk is chopped in synchronization with the input signal (Fig. 3.6).

3.2. System Transfer function

This section discusses the overall transfer function of the system, assuming the EDO has already been compensated by the proposed Bi-Level EDO compensation loop.

3.2.1. Gain of the system

The system gain is defined by the boxcar sampler. The ideal gain of the boxcar sampling is shown in Equation 3.2.

$$A_v = \frac{V_{\text{OUT}}(t)}{V_{\text{IN}}(t)} = \frac{g_m T_{\text{INT}}}{C_{\text{INT}}} \tag{3.2}$$

This equation represents the gain of the boxcar sampler in an ideal scenario, where g_m is the transconductance, T_{INT} is the integration time, and C_{INT} is the integrating capacitor. In this ideal operation, the output impedance r_o of the OTA is considered infinite. However, in practical applications, the output impedance r_o of the OTA has a finite value. Considering an OTA with a limited output impedance

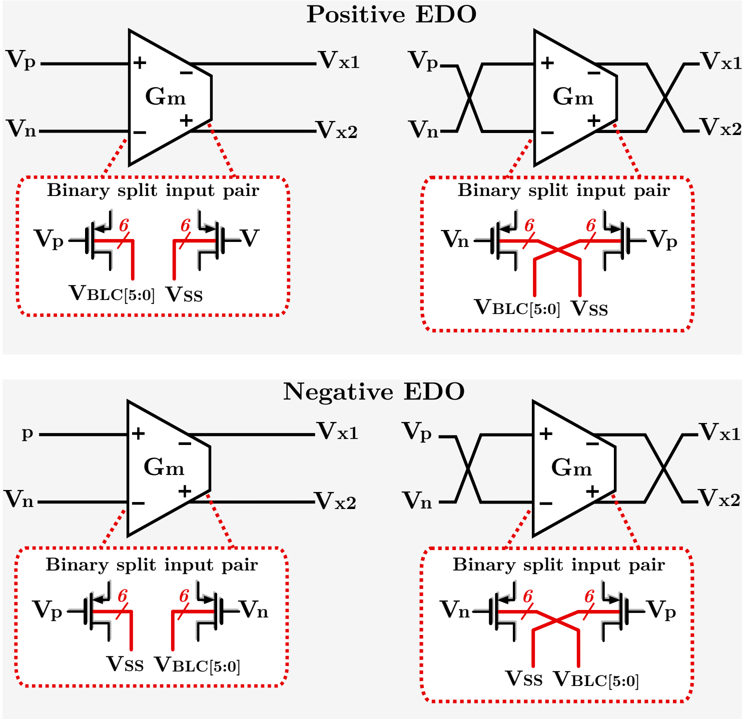


Figure 3.6: Connections during both phases of the chopper stabilization with positive or negative EDOs (with the BLC connection highlighted in red).

(r_o), the gain of the boxcar sampler is derived as shown in Equation 3.3.

$$A_v = \frac{V_{\text{OUT}}(t)}{V_{\text{IN}}(t)} = g_m r_o \left(1 - e^{-T_{\text{INT}}/r_o C_{\text{INT}}}\right) \tag{3.3}$$

From this equation, it can be seen that the transfer function of the boxcar sampler exhibits an exponential decay characteristic. To maintain the sinc function along with the introduced notches across the multiples of the sampling frequency (as explained in subsection 2.3.3), the time constant formed by the OTA output impedance and the integrating capacitor ($\tau = r_o C_{\text{INT}}$) must be large enough so the gain is not fully settled at the instant of T_{INT} ($T_{\text{INT}} \ll \tau$).

If $T_{\text{INT}} \ll \tau$, the gain of the boxcar can be approximated as shown in Equation 3.2.

Figure 3.7 provides an example of the effective gain of a boxcar sampler, comparing the ideal gain with a case where the τ is large enough ($\tau = 20 \cdot T_{INT}$) and the case where this time constant is considerably smaller ($\tau = 0.2 \cdot T_{INT}$).

When τ is large enough (blue), the boxcar gain versus time function exhibits a linear response that can be approximated as the ideal case, with a small gain error. However, if τ is smaller than desired (red), the response is a decaying exponential function. Consequently, the transfer function no longer exhibits the characteristics of a boxcar sampler.

3.2.2. Anti-aliasing and sampling

As discussed in subsection 2.3.3, boxcar sampling offers improved conditions in terms of compactness. It combines both the anti-aliasing and sampling phases, and it effectively minimizes noise folding. In addition, the attenuation notches introduced by the equivalent sinc function are used to attenuate the

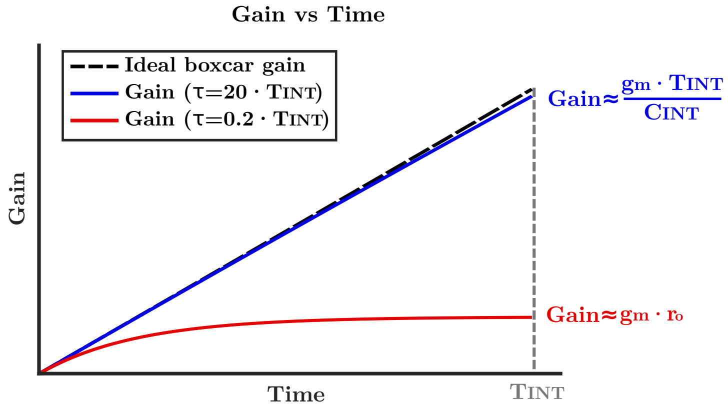


Figure 3.7: Comparison of the gain of the boxcar sampler depending on the time constant of the circuit.

high-frequency chopping when f_{chop} is chosen to be a multiple of f_s.

The location of the low-pass filter pole introduced by the boxcar sampling follows the equations Equation 3.4 and Equation 3.5.

$$\frac{\sin\left(\mathrm{T_{INT}f}\right)}{\mathrm{T_{INT}f}} = \frac{1}{\sqrt{2}} \tag{3.4}$$

$$\mathrm{f_{-3dB,\ boxcar}} = \frac{0.443}{\mathrm{T_{INT}}} \tag{3.5}$$

where $\mathrm{T_{INT}}$ corresponds to the integration time. This expression indicates that achieving a $\mathrm{f_{-3dB,\ boxcar}}$ at 500 Hz (to act as an anti-aliasing filter and limit the bandwidth of interest), requires an integration time of 0.886 ms. This value implies a theoretical maximum sampling rate, f_s, of 1128 Hz (considering the resetting period, T_{RST}, of the boxcar negligible).

However, the ADC comparator requires a minimum conversion time, $T_{\mathrm{conversion}}$. This conversion must occur after T_{INT} and before T_{RST} (as a static value in C_{INT} is needed for comparison). Ensuring sufficient time for T_{RST}, $T_{\mathrm{conversion}}$, and the defined T_{INT} while also fulfilling the Nyquist-Shannon theorem and achieving a sampling rate above 1000 Hz (twice the bandwidth of the signal) is challenging.

To alleviate this design constraint, an additional sampling capacitor, C_{LPF}, is introduced. With this approach, the amplified, sampled and filtered signal (after the boxcar sampling) is sampled again to C_{LPF}, allowing the comparator to perform its comparison for a full sampling period T_{s} (accounting for the tracking time associated with sampling onto C_{LPF}, T_{LPF}). These two alternatives are conceptually compared in Figure 3.8.

The additional sampling phase adds latency to the BLC. When one of the thresholds is crossed, the BLC compensation affects the following integration phase. However, its effect is not seen in the immediate ADC output, but in the subsequent one (Figure 3.9). This latency in the loop could potentially destabilize the system if the crossing of thresholds were evaluated after every sample, as it might compensate for an EDO that has already been addressed. To mitigate this risk, the comparison of D_{ADC} with the digital thresholds is performed in an alternating pattern: one sample is evaluated, the next is skipped, then the following is evaluated, and so on.

Moreover, both the C_{INT} and C_{LPF} are reused as a SC-LPF (previously introduced in subsection 2.3.1). With this approach, C_{INT} acts as an equivalent resistor (Fig. 3.10).

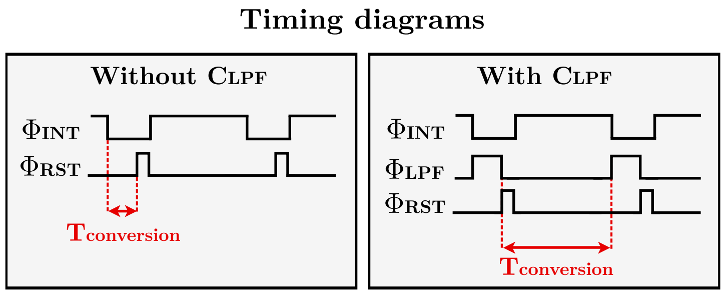


Figure 3.8: Comparison of the available time for the ADC conversion ($T_{conversion}$) with and without the additional sampling capacitor (C_{LPF}).

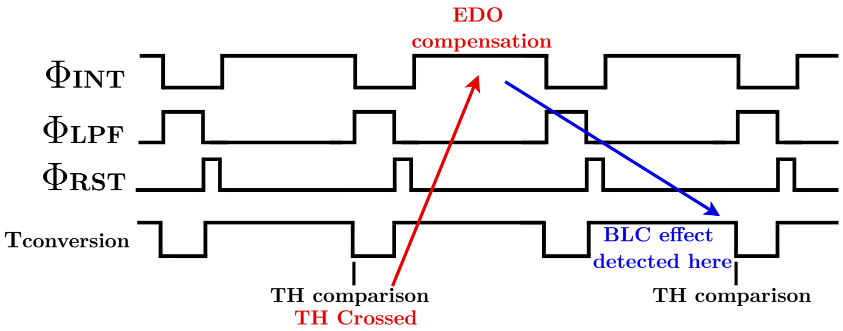


Figure 3.9: Introduced BLC latency.

The equivalent resistor value of the switched C_{INT} is derived in Equation 3.6.

$$\mathrm{R_{eq}} = \frac{\mathrm{T_{INT} + T_{LPF}}}{\mathrm{C_{INT}}} \tag{3.6}$$

Consequently, the LPF pole is set as shown in Equation 3.7:

$$\mathrm{f_{SC-LPF}} = \frac{1}{2\pi \mathrm{R_{eq} C_{LPF}}} = \frac{\mathrm{C_{INT}}}{2\pi \mathrm{C_{LPF}}} \frac{1}{\mathrm{T_{INT} + T_{LPF}}} \tag{3.7}$$

The following expression (eq. 3.8) to set the ratio between the two capacitors and achieve the desired LPF cut-off frequency (f_{-3dB}).

$$\frac{\sin\left(\mathrm{T_{INT} f_{-3\ dB}}\right)}{\mathrm{T_{INT} f_{-3\ dB}}} \cdot \frac{1}{\sqrt{1 + \left(2\pi \mathrm{f_{-3\ dB}}\left(\,\mathrm{T_{INT} + T_{LPF}}\right) \frac{\mathrm{C_{LPF}}}{\mathrm{C_{INT}}}\right)^2}} = \frac{1}{\sqrt{2}} \tag{3.8}$$

In the worst case (if the C_{LPF} is uncharged), the boxcar sampler will be attenuated approximately by the ratio of the two capacitors due to charge redistribution, following (Equation 3.9).

$$A_{v,SC} = 1 - \frac{C_{LPF}}{C_{\mathrm{INT}} \ + C_{\mathrm{LPF}}} \tag{3.9}$$

However, as C_{LPF} is never reset, it retains the charge from the previous sample, resulting in negligible attenuation for low frequencies, such as the signal of interest.

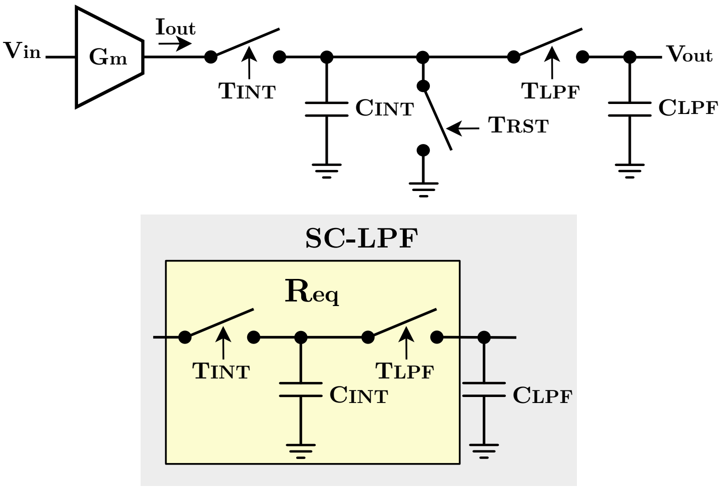


Figure 3.10: SCF implemented with boxcar sampler integration capacitor (C_{INT}) and an additional sampling capacitor (C_{LPF}).

3.2.3. SS-ADC

The last element in this signal acquisition chain is the ADC, which converts this sampled analog value to a digital code. The implemented SS-ADC has the benefit of reducing the ADC to a single comparator for each of the recording AFE while a differential ramp generated by a global ramp generator is distributed to all the pixels.

The number of bits for the ADC typically depends on the specific application for which the AFE is designed. However, this research does not target a single specific application but rather a range of neural signal recording scenarios. To select an appropriate resolution for this generalized case, typical neural signal characteristics are considered. The neural signal amplitude generally ranges between 10 μV_{pp} and 1 mV_{pp} (as exposed in subsection 1.1.2), while the AFE noise level is limited to a maximum value of 2 μV_{rms} (justified in Equation 1.1.4). Based on these typical conditions, the maximum achievable SNR is around 45 dB. Therefore, for this generalized neural signal recording scenario, a SS-ADC with 8-bit resolution is theoretically sufficient to capture the range of signals of interest.

However, the effective bit count allocated to these signals will be lower due to the requirement of also digitizing the residual EDO (as explained in subsection 3.1.1). Therefore, it is essential to increase the system's bit count to provide additional redundancy while maintaining the signal integrity.

Moreover, as exposed in the previous sections, the gain of the system is only determined by T_{INT} and C_{INT} as the g_m is fixed by the thermal noise requirements (detailed in section 3.3). Then, to limit the gain of this stage without requiring an impractically large integrating capacitor, T_{INT} can be decreased. However, decreasing the integration time entails worse noise folding as it results in a higher $\text{f}_{\text{-3dB, boxcar}}$ (eq. 3.5). Oversampling the ADC can compensate for this extra noise, and increase the overall number of bits in order to provide the required redundancy. As discussed on subsection 2.4.1, oversampling improves the SNR of the ADC and provides an additional bit of resolution for every time the OSR is increased by 4x.

Accounting for these requirements, the sampling frequency (f_s) is chosen to be 16 times higher than the Nyquist frequency of the system (f_{nyquist}), entailing an OSR of 16 and theoretically providing additional 4 bits of resolution. To achieve this, an accumulator is used to sum each 8-bit output code of the ADC up to 16 times. The result of this operation is a 12-bit code at the Nyquist rate. After each accumulation operation, the accumulator register is reset to start the next cycle. The principle of this operation is described as in Equation 3.10.

$$y[k] = \frac{1}{16}\sum_{n=0}^{15} x[k \cdot 16 + n] \tag{3.10}$$

here $y[k]$ represents the output code at the Nyquist rate, k is the index of the Nyquist rate output. The term $x[n]$ represents the 8-bit output code from the ADC with n being the index of the output at the oversampling rate.

This procedure is similar to a Finite Impulse Response (FIR) filter with all coefficients having equal weights, also known as a rectangular window filter. Although other window functions with different weight distributions could be explored, they require more complex implementations and therefore are not considered [122], [123].

The impulse response of the implemented filter $h[n]$ is defined by:

$$h[n] = \begin{cases} \frac{1}{\text{OSR}} & \text{for } 0 \leq n < \text{OSR} \\ 0 & \text{otherwise} \end{cases} \tag{3.11}$$

This indicates a uniform filter where the coefficient is $\frac{1}{\text{OSR}}$ over OSR 8-bit ADC output codes.

The frequency response of this filter can then be derived with the Fourier transform of the impulse response. Given that the impulse response is a simple constant over the OSR points, the frequency response $H(f)$ is given by:

$$H(f) = \frac{1}{\text{OSR}} \cdot \frac{\sin\left(\frac{\pi f \text{OSR}}{f_s}\right)}{\left(\frac{\pi f}{f_s}\right)} \cdot e^{-j\frac{\pi f(\text{OSR}-1)}{f_s}} \tag{3.12}$$

Therefore, the magnitude transfer function is derived as:

$$|H(f)| = \frac{1}{\text{OSR}} \cdot \frac{\left|\sin\left(\frac{\pi f \text{OSR}}{f_s}\right)\right|}{\left|\left(\frac{\pi f}{f_s}\right)\right|} \tag{3.13}$$

This introduces a sinc-like envelope, resulting in nulls at integer multiples of $\frac{f_s}{\text{OSR}}$ and a set of side-lobes that initially roll off at -20 dB/decade.

3.3. System noise analysis

To ensure that the neural signals are recorded with sufficient fidelity, it is crucial to consider the noise performance of the entire system. Figure 3.11 illustrates the noise sources present in the Analog Front End (AFE) system, highlighted in red. The noise generated by the ADC is not included in this analysis.

3.3.1. OTA

The noise contribution of the OTA is directly referred to the input and is highly dependent on the circuit topology. Therefore, as a preliminary analysis, the noise of the OTA is assumed to be equivalent

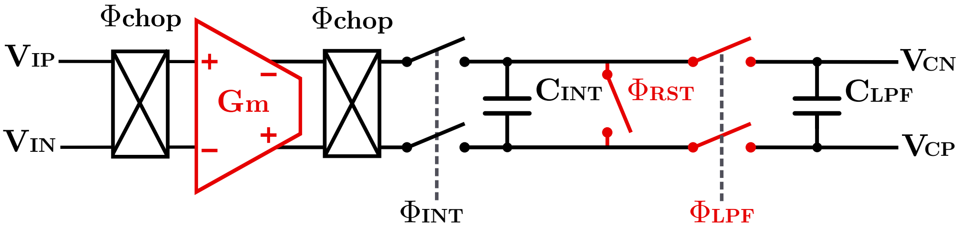


Figure 3.11: Components of the system architecture considered in the noise analysis (highlighted in red).

to that of an input pair of transistors, due to its differential architecture. Additionally, since chopping modulation is implemented, only thermal noise is considered significant, while flicker noise is considered negligible. The input referred noise power spectral density (PSD) of the OTA is defined in Equation 3.14.

$$V^2_{n,\text{in, OTA}} = \frac{8k_BT\gamma}{g_{m,\text{OTA}}}\ [\text{V}^2\ \text{Hz}^{-1}] \tag{3.14}$$

3.3.2. Boxcar sampling and SCF-LPF circuits

In every sampling cycle, kT/C noise (explained in subsection 2.3.2) is introduced by the resistance of the sampling switches. The magnitude of the kT/C noise introduced by the boxcar sampler is exposed in Equation 3.15.

$$V^2_{n,\text{rms,BS}} = \frac{k_BT}{C_{\text{INT}}}\ [\text{V}^2] \tag{3.15}$$

The sampling voltage noise from the boxcar sampler is attenuated approximately by a factor of $\frac{\text{C}_{\text{INT}}}{\text{C}_{\text{INT}}+\text{C}_{\text{LPF}}}$due to the charge redistribution between C_{INT} and C_{LPF}.

The sampling noise of the second sampling circuit (also used as a passive LPF) is derived as:

$$V^2_{n,\text{rms,LPF}} = \frac{k_BT}{C_{\text{LPF}}}\ [\text{V}^2] \tag{3.16}$$

3.3.3. Additional noise sources

The comparator used in the SS-ADC introduces both thermal and flicker noise into the system. However, deriving this noise is not straightforward because the comparator is a non-linear component. Additionally, the noise from the comparator is significantly attenuated by the gain of the Boxcar sampler when referred to the input, making it negligible. For these reasons, it is not considered in this analysis.

The switches employed in the chopping modulation or the Bi-level EDO compensation loop also introduce thermal noise proportional to their on-resistance. However, since this noise depends on their specific circuit implementation, it is not considered in this section and is explored in chapter 4.

3.3.4. Overall system input referred noise derivation

The input referred noise for the entire system is derived by Equation 3.17.

$$V_{\mathrm{n,rms,in}}^2 = V_{\mathrm{n,OTA}}^2 \cdot \mathrm{BW} + \frac{2 \cdot V_{\mathrm{n,rms,BS}}^2 \cdot \left(\frac{C_{\mathrm{INT}}}{C_{\mathrm{INT}}+C_{\mathrm{LPF}}}\right)^2}{A_{\mathrm{v}}^2} + \frac{2 \cdot V_{\mathrm{n,rms,LPF}}^2}{A_{\mathrm{v}}^2} \tag{3.17}$$

Where BW is the system bandwidth and A_v is the voltage gain at the output of the boxcar sampler. From this previous expression, it can be seen that the dominant noise contributor of the entire system is the OTA (as all the other sources are attenuated by the boxcar gain when referring to the input).

3.4. Preliminary design

The total transfer function of the system, including the effect of the boxcar sampler, passive SC-LPF, and the accumulation operation of the oversampling ADC, can be seen in Figure 3.12.

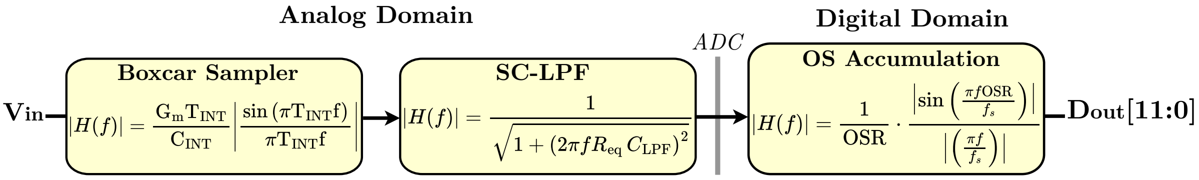

Analog Domain
Digital Domain
ADC
Vin
Boxcar Sampler
$|H(f)| = \frac{\mathrm{G_m T_{INT}}}{\mathrm{C_{INT}}} \left| \frac{\sin(\pi \mathrm{T_{INT}} f)}{\pi \mathrm{T_{INT}} f} \right|$
SC-LPF
$|H(f)| = \frac{1}{\sqrt{1 + (2\pi f R_{\mathrm{eq}} C_{\mathrm{LPF}})^2}}$
OS Accumulation
$|H(f)| = \frac{1}{\mathrm{OSR}} \cdot \frac{\left|\sin\left(\frac{\pi f \mathrm{OSR}}{f_s}\right)\right|}{\left|\left(\frac{\pi f}{f_s}\right)\right|}$
Dout[11:0]

Figure 3.12: System equivalent using the transfer function of each block.

To obtain an initial theoretical estimate of the system-level response for the proposed AFE, preliminary values for the parameters defining the AFE are derived.

OTA

To make a preliminary choice of the required OTA g_m value based on Equation 3.14 and the noise requirement of $2\,\mu\mathrm{V_{rms}}$ presented in Table 1.1, the following assumptions are considered:

- γ is assumed to be 1.
- The noise is integrated over a bandwidth of 500 Hz, which is the Equivalent Noise Bandwidth (ENBW) due to the use of the LNB on this architecture.
- The temperature is assumed to be 300 K.
- The OTA is implemented with a folded cascode topology (explained in chapter 4) and the input pair of the OTA can be approximated to contribute 65% of the total noise in the AFE. This is because:
  - The OTA is the main noise source in the AFE.
  - Other transistors in the OTA topology also contribute to noise (assumed to be around 15% of the noise contribution).
  - Other noise sources mentioned in section 3.3 such as the kT/C noise introduced by the SC-LPF will add to the total noise. For this approximation, 15% of the noise is attributed to these sources.
  - Chopping stabilization might not completely eliminate flicker noise in efforts to reduce the chopping frequency (assumed to be 5% of the noise contribution).

With these considerations, the first approximation of the minimum g_m can be derived as in Equation 3.18.

$$g_{m,\mathrm{ota}} > \frac{8 \cdot K_b \cdot \gamma \cdot BW}{2\mu V} \tag{3.18}$$

From the previous equation, a minimum g_m of 9.57 μS is obtained. Therefore for first-order approximation of the OTA a $g_{m,\text{ota}}$ of 10 μS is chosen.

Boxcar sampler and SC-LPF

Setting a sampling frequency of 16 kHz (considering the previously mentioned OSR ratio) also determines the maximum T_{INT}. Sufficient time must be allocated for the T_{RST} and T_{LPF} phases. As a first approximation, T_{INT} is assumed to be up to 99% of the sampling period, leading to a $T_{INT,approx}$ of 61.25 μs.

The capacitor size can be derived from Equation 3.19 for a chosen gain Δ_v.

$$C_{\text{INT}} = \frac{g_{m,\text{ota}} \cdot T_{\text{INT, approx}}}{\Delta_v} \tag{3.19}$$

Due to the small amplitude of the recorded neural signals, as specified in subsection 1.1.2, it is necessary to employ a high differential gain (greater than 40 dB) [37], [38], [124]. This approach relaxes the dynamic range requirements of the subsequent analog-to-digital converter (ADC) and minimizes the noise contribution from later stages, since their noise, when referred to the input, is reduced, as demonstrated in Equation 3.17. However, to prevent the LNB from saturating in the presence of a residual EDO at the input, the maximum gain is limited at 46 dB.

To achieve a gain between 100 V/V and 200 V/V, the capacitor value $C_{\text{INT, approx}}$ must be between 3 pF and 6 pF. While increasing the capacitor value reduces kT/C noise, it also decreases the OTA gain, which then increases input-referred noise contribution, as shown in Equation 3.17. Additionally, a higher C_{INT} value reduces the equivalent resistance of the SC-LPF, thereby increasing the f_{-3dB} leading to a worse attenuation of high-frequency noise components (see eq. 3.7). Considering these limitations and aiming to minimize the implementation area, C_{INT} is chosen to be 4 pF leading to a theoretical ideal gain of 150 V/V.

To select a value for C_{LPF}, equation Equation 3.8 is considered. Assuming T_{INT} of 61.25 μs and T_{LPF} of 0.625 μs, a ratio of $\frac{C_{\text{LPF}}}{C_{\text{INT}}} = 5.14$ is required to set $f_{\text{SC-LPF}}$ to 500 Hz. With the previously chosen $C_{\text{INT, approx}}$ of 4 pF, this would necessitate a C_{LPF} of 20.56 pF. However, such a value leads to significant area overhead.

To choose an appropriate value for this capacitor, it is necessary to balance the filter bandwidth, kT/C noise (as shown in Equation 3.16), and area overhead. The SC-LPF's overall frequency response is influenced by two factors: its first-order low-pass transfer function (see Fig. 3.12) and the inherent sinc effect arising from the discrete-time sampling nature of the switched-capacitor circuit. A more accurate representation of the SC-LPF frequency response is:

$$|H(f)| = \left| \left(\frac{\sin\left(\pi \frac{f}{f_{SC}}\right)}{\pi \frac{f}{f_{SC}}} \right) \right| \cdot H_{\text{LPF}}(f) \tag{3.20}$$

Where f_{SC} is the SC-LPF switching frequency $(1/(T_{INT} + T_{LPF}))$, and $H_{\text{LPF}}(f)$ is the first-order low-pass filter frequency response. This sinc term introduces notches at multiples of the switching frequency, providing additional high-frequency attenuation and enhancing the filter's low-pass characteristics.

Notably, this sinc effect is less apparent when the sampling frequency (f_{SC}) significantly exceeds the filter's cutoff frequency ($f_{-3\text{dB}}$) - the typical design approach. In this design, however, this effect is advantageous as it further attenuates high-frequency noise.

Considering this sinc effect and the additional bandwidth limitation from the accumulation operation's low-pass filter (detailed in subsection 3.2.3), the SC-LPF pole location requirements can be relaxed. This relaxation allows for a smaller C_{LPF} without compromising high-frequency noise rejection. Consequently, a $C_{\mathrm{LPF}} = 0.5\,\mathrm{pF}$ is selected. Although this smaller capacitor shifts the SC-LPF pole to a higher frequency (20.7 kHz), the combined effect of the sinc response and subsequent filtering stages ensures sufficient high-frequency attenuation. Moreover, the input referred noise contributions for C_{INT} and C_{LPF} (considering a boxcar gain of 150 V/V), are 285 nV_{rms} and 0.85 μV_{rms} respectively, which is considerably below the aimed noise level.

Theoretical estimation of the system-level response for the proposed AFE

The system-level response for the first approximation of parameters (Table 3.1) is shown in figures 3.13, 3.14 and 3.15.

From Figure 3.13, it is possible to identify a gain of 150 V/V (43.5 dB) and the introduction of sinc notches at multiples of T_{INT}.

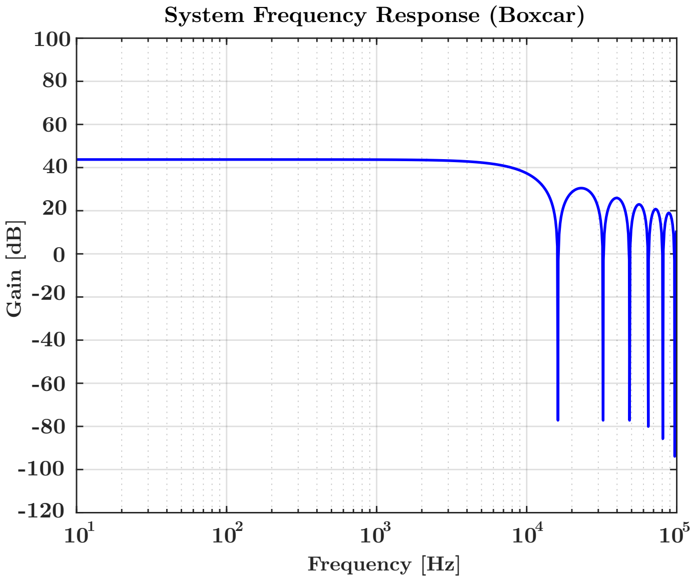


Figure 3.13: Transfer function of the boxcar (with the chosen values).

Figure 3.14, illustrates the further attenuation of the high frequencies introduced by the SC-LPF.

Figure 3.15 shows the effect of the digital accumulator on top of the analog Boxcar and SC-LPF. The digital accumulator introduces notches at multiples of 1000 kHz, resulting in a -3 dB bandwidth of approximately 500 Hz (but slightly lower). To achieve the precise target -3 dB bandwidth of 500 Hz while maintaining the 43.5 dB gain, potential improvements could include slightly increasing the sampling frequency to shift the sinc notches higher in frequency, or implementing a counter filter before the first notch.

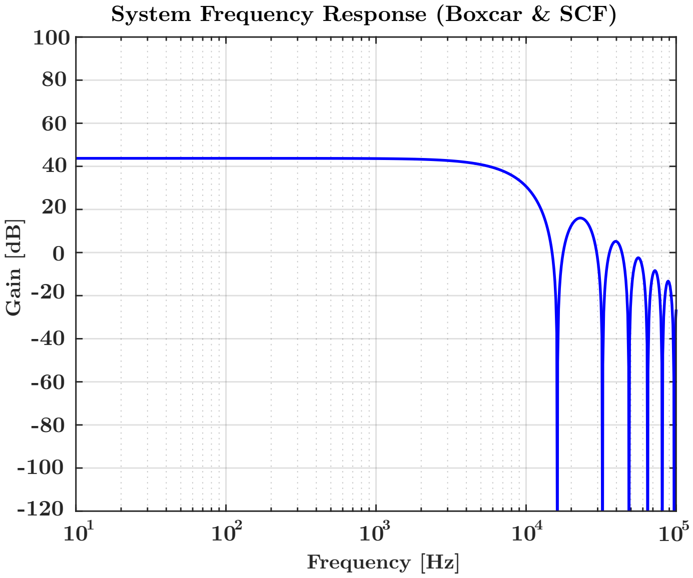


Figure 3.14: Transfer function of the boxcar and SC-LPF (with the chosen values).

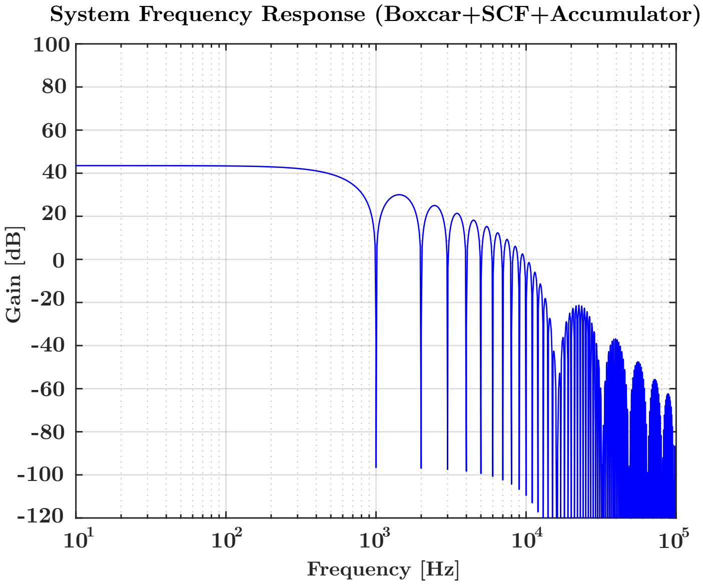


Figure 3.15: Transfer function of the boxcar, SC-LPF, and digital accumulator (with the chosen values).

Table 3.1: Design parameters used to simulate the first approximation of the AFE transfer function.

Parameter	Value	Justification
g_m	10 μS	To meet the input-referred noise requirement.
C_{INT}	4 pF	To meet the desired gain of the boxcar sampler and to implement the SC-LPF.
C_{LPF}	0.5 pF	To implement the SC-LPF and meet the abovementioned constraints.
T_{INT}	61.25 μs	To meet desired gain of the boxcar sampler.

4

Circuit level design

In this chapter, the circuit-level design of the proposed AFE (Fig. 3.1) is presented and discussed. This design is implemented using TSMC 40 nm standard CMOS technology with a voltage supply of $V_{DD} = 1.1$ V.

4.1. Boxcar sampler

The circuit level implementation of the boxcar sampler is divided into two main components: the OTA and the switching and sampling components including the passive SC-LPF.

4.1.1. OTA

In neural recording applications, selecting the optimal amplifier topology is crucial for achieving low-power and low-noise performance. Among the various configurations, inverter-based OTAs and folded cascode amplifiers are the most common choices [89].

Inverter-based OTAs provide twice the transconductance for the same amount of current, effectively reducing the IRN with half the power consumption. Additionally, this architecture is composed of only six transistors, making it highly desirable in terms of area efficiency. However, inverter-based OTAs exhibit limitations, such as a lower CMRR and a smaller output impedance. Moreover, the OTA used in the boxcar sampler must employ only PMOS devices as input pair to freely control the body connection within a safe operating zone. Due to these requirements, the folded cascode topology is chosen for its better CMRR (required as stated in Table 1.1) and higher output impedance which will result in a smaller gain error (as explained in subsection 3.2.1). In addition, a fully-differential architecture is chosen for further increased CMRR. Figure 4.1 illustrates the schematic of the fully-differential folded cascode OTA.

The gain of the amplifier is

$$|A_v| = g_{m1} \cdot r_{ota} = g_{m1} \left\{ \left[\left(g_{m3} + g_{mb3} \right) r_{o3} \left(r_{o1} \| r_{o2} \right) \right] \| \left[\left(g_{m4} + g_{mb4} \right) r_{o4} r_{o5} \right] \right\} \tag{4.1}$$

where g_m is the gate transconductance, g_{mb} is the body transconductance, and r_o is the output impedance, with each of these parameters specific to the respective transistor.

The output swing present in the output branch can be derived as shown in Equation 4.2

$$V_{out,swing} = V_{DD} - V_{ds2} - V_{dsat3} - V_{dsat4} - V_{ds5} \tag{4.2}$$

The thermal noise contribution of the folded cascode (considering the contribution of the cascode tran-

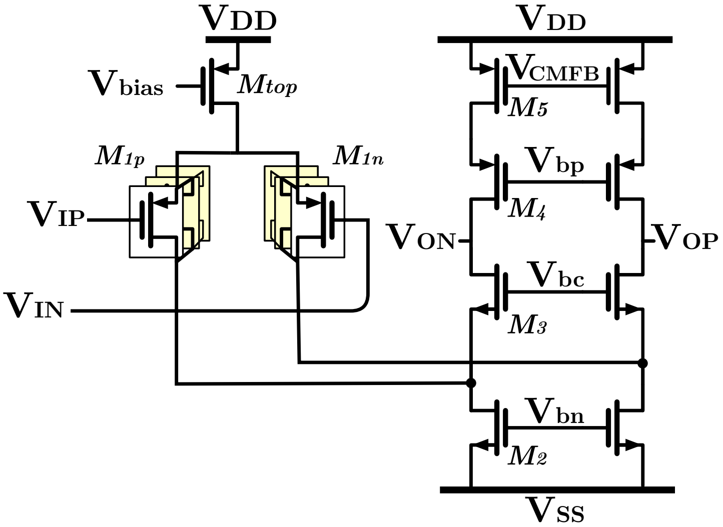


Figure 4.1: Implementation of the folded cascode OTA with split input pair.

sistors negligible) is shown in Equation 4.3

$$\overline{V_{n,int}^2} = 8kT\gamma\left(\frac{1}{g_{m1}} + \frac{g_{m2}}{g_{m1}^2} + \frac{g_{m5}}{g_{m1}^2}\right) \tag{4.3}$$

Additionally, the flicker noise corner must be controlled to set the chopping frequency optimally. The flicker noise corner follows the expression in Equation 4.3

$$f_{co} = \frac{K_f}{C_{ox}}\frac{g_m}{I_D}\frac{I_D}{WL}\frac{1}{4k_BT\gamma} \tag{4.4}$$

Where K_f is a process parameter that depends on the W and L of a transistor and C_{ox} is the oxide capacitance per unit area of the used CMOS technology node.

Lastly, achieving a high CMRR is crucial for the OTA due to the prevalence of common-mode interference. To ensure this, M_{top} must have a high output impedance to prevent common-mode noise from affecting the bias current. In addition, good matching between the OTA transistors ensures consistent processing of the desired signal (differential mode) on both sides of the amplifier. However, if there is inadequate matching between these transistors, it can lead to common-mode to differential conversion, where common-mode noise is unintentionally transformed into a differential signal. This conversion significantly degrades the CMRR of the amplifier, and consequently of the entire AFE.

This section presents the procedure used for sizing the circuit accounting for the abovementioned parameters. To proceed with a systematic approach rather than relying on performing multiple sweeping operations to select the adequate value, the method present in [125] is chosen instead. This method uses pre-computed lookup tables that employ the trans-conductance efficiency (gm/ID) as a proxy and key parameter for the design. The Matlab code employed for this purpose is provided in Appendix A.

Biasing current

The first step of the design is to choose the biasing current for both the input and output branches to meet the noise, power, and swing requirements. The voltage-to-current conversion primarily relies on the input transistors, which must ensure low noise performance. In contrast, the output stage mainly serves to increase the architecture's output impedance. This allows the output branch to be biased with only a small portion of the current used for the input transistors, contributing to power efficiency. Using a smaller bias current in the output branch can potentially limit the output swing. This limitation arises because the input current cannot be fully steered to a single branch if the output current is significantly smaller. However, due to the required small input swing, this limitation is not detrimental for this design.

The overall noise of the OTA (eq. 4.3) is mainly contributed by the input transistors (M_{1p} and M_{1n}) and the tail transistors (M_2), as their drain current entails the summation of the input biasing and output branch biasing current. Therefore, the thermal noise contribution can be approximated as shown in Equation 4.5.

$$\overline{V_{n,int}^2} \approx 8kT\gamma\left(\frac{1}{g_{m1}} + \frac{g_{m2}}{g_{m1}^2}\right) \tag{4.5}$$

In strong inversion, γ is typically considered to be $\frac{2}{3}$, whereas in the subthreshold condition, it is closer to 1. For all subsequent calculations, a value of 1 is assigned to γ for all transistors to provide a safety margin.

Thick oxide devices are chosen for the input transistors to minimize gate leakage and, therefore, the corresponding shot noise as in [29]. In addition, a GM/ID ratio of 25 is chosen to bias these transistors in weak inversion and maximize the noise efficiency of the OTA. This choice leads to a $V_{dsat} \approx \frac{2}{g_m/I_D} =$ 80mV.

The bottom transistors of the output branch (M_2) should be biased in strong inversion to minimize their g_m and consequently their input-referred noise (eq. 4.5). Biasing them in strong inversion also contributes to better threshold matching with the current mirrors used for biasing (further explained in Equation 4.10). However, a large output swing is needed to accommodate the residual DC offset and the input signal without distortion, which would require a lower V_{dsat}. Therefore, as a compromise between these requirements, a GM/ID ratio of 15 is chosen to bias them in moderate inversion. The corresponding saturation voltage for these transistors is then: $V_{dsat} \approx \frac{2}{g_m/I_D} = 133\text{mV}$.

Then, to have an input-referred noise contribution of approximately $1.7\,\mu\text{V}_{\text{rms}}$ (below the required $2\,\mu\text{V}_{\text{rms}}$ to ensure adequate noise headroom for other noise contributors), an input biasing current ($I_{\text{OTA,in}}$) of 800 nA is chosen. The biasing current for the output branch ($I_{\text{OTA,out}}$) is chosen to be 10 times smaller, 80 nA. The corresponding g_m of the OTA is then $10\,\mu$S.

Input branch sizing

To determine the value for the width and length of the input transistors with the previously obtained g_m, is important to consider which V_{DS} will be allowed for the input transistors to set them in deep saturation while allowing sufficient headroom voltage for M_{top}. With a DC voltage applied at the gate of the input transistors of 0V (as further explained in section subsection 4.1.2), the V_{DS} voltage of the input current source is derived as Equation 4.6.

$$|V_{DS,topcs}| = V_{DD} - |V_{GS,in}| \tag{4.6}$$

In addition, the drain to source voltage of the input transistor can be derived as shown in Equation 4.7.

$$|V_{DS,input}| = V_{GS,in} - V_{ds,2} \tag{4.7}$$

On the other hand, as the input transistors are binary split as part of the Bi-level compensation

(BLC) implementation, it is necessary to ensure that the switching scheme is monotonic and resilient to mismatch. The mismatches in current and threshold of the transistors are modeled as [126].

$$\sigma^2_{\Delta\beta|\beta} = \frac{A^2_\beta}{WL}, \tag{4.8}$$

$$\sigma^2_{\Delta V_T} = \frac{A^2_{VT}}{WL} \tag{4.9}$$

Where the matching parameters A_β and A_{VT} are technology dependent. From the previous equations, it is shown that the mismatch variances scale inversely with the device area. Therefore, larger transistors are advantageous as they reduce mismatch variance. Additionally, increasing the length of the transistors will decrease the 1/f noise corner frequency, thereby allowing for a lower chopping frequency (f_{chop}) as illustrated in Figure 4.2. However, this also increases the parasitic capacitance present at the input.

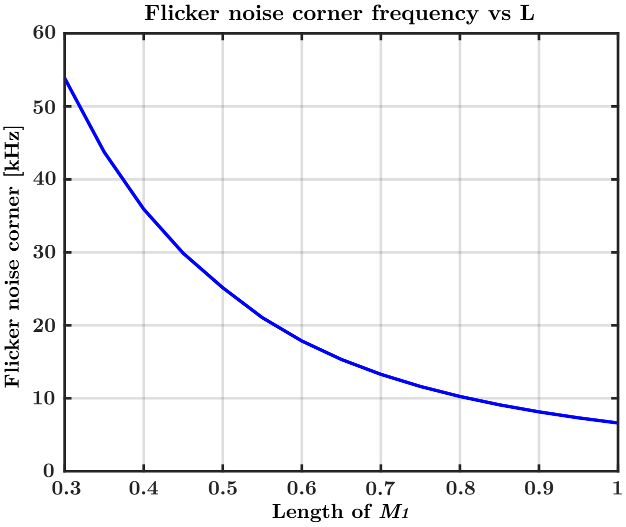


Figure 4.2: Flicker noise corner approximation of the input transistors vs their length.

As a compromise, $W_{in}/L_{in} = \frac{26\mu m}{1\mu m}$ is chosen. Next, for the sizing of M_{top}, the effect of mismatch in current sources must be considered. The mismatch in current sources can be extended from Equation 4.8 and Equation 4.9 and derived as:

$$\sigma^2_{\Delta I_D/I_{D1}} \cong \left(\frac{g_{m1}}{I_{D1}}\right)^2 \sigma^2_{\Delta V_T} + \sigma^2_{\Delta\beta/\beta}. \tag{4.10}$$

Therefore, to achieve better threshold matching of the top current source (M_{top}) with the biasing circuit (explained in subsection 4.1.2), a GM/ID of 7 is chosen for this device to bias it in strong inversion.

Additionally, as mentioned, the output impedance of the current source is directly proportional to the CMRR of the amplifier. The output impedance of the device increases proportionally with its length. Moreover, larger devices reduce the mismatch error within the biasing circuit. For these reasons, a $W_{top}/L_{top} = \frac{1\mu m}{4\mu m}$ is selected.

Output branch sizing

Considering the tradeoff between matching, noise, and output swing, the decision is to choose a GM/ID of 15 for the bias devices (M_2). Additionally, the top current sources (M_5) are also biased with the same GM/ID ratio, resulting in $V_{\mathrm{dsat}} \approx 133\,\mathrm{mV}$. By allocating a drain-source voltage of 200 mV (explained in subsection 4.1.2), both devices are given 67 mV of headroom to operate in saturation.

In order to optimize the output swing, the V_{dsat} of the cascode devices (M_3, and M_4) must be reduced (requiring a higher GM/ID). On the other hand, increasing this ratio will also increase the intrinsic capacitances of these devices, loading the output of the OTA and reducing the gain. As a compromise, a GM/ID of 20 is chosen for these devices, leading to a $V_{\mathrm{dsat}} \approx 100\mathrm{mV}$.

In order to choose the sizes of the output branch devices, it is essential to consider the primary purpose of the output branch, which is to provide a higher output impedance. Therefore, it is important to evaluate the effect of the chosen lengths for these devices and how they affect the r_{out}. Figure 4.3 presents the obtained output impedance with different transistor lengths (while keeping the same length for all transistors in the output branch).

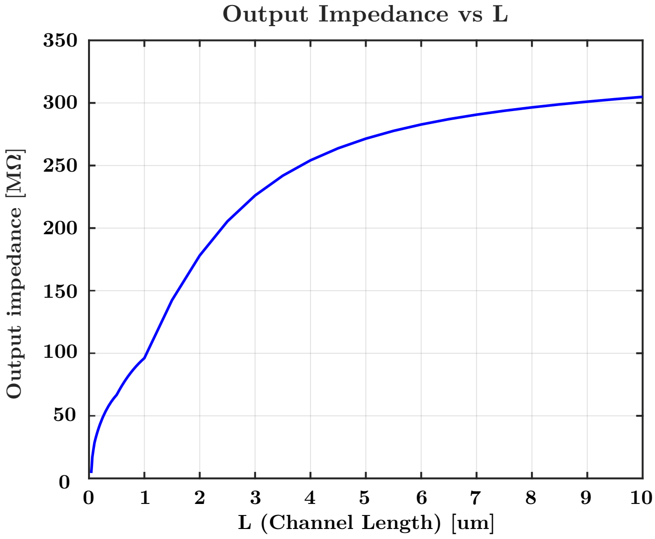


Figure 4.3: Output impedance of the OTA vs the length of the output branch transistors.

The value of the output impedance has a direct impact on the boxcar gain amplification, contributing to a gain error (explained in subsection 3.2.1). The resulting gain error vs the length of the transistors is exposed in Figure 4.4.

Additionally, it is necessary to consider that the bottom and top transistors are the main noise contributors of the output branch. Even though the thermal noise of these transistors has already been accounted for in the previous noise calculations, the $1/f$ noise corner of the amplifier is also influenced by these transistors (Fig. 4.5).

Considering the abovementioned factors, a length of $4\,\mu$m is chosen for all the output branch transistors. This choice is made to achieve the following objectives: minimize the gain error from boxcar sampling, minimize the mismatch of the output branch transistors (which decreases with larger lengths as explained in Equation 4.8 and Equation 4.9), and reduce the $1/f$ corner frequency. In addition, having the same lengths for all these transistors simplifies the layout process.

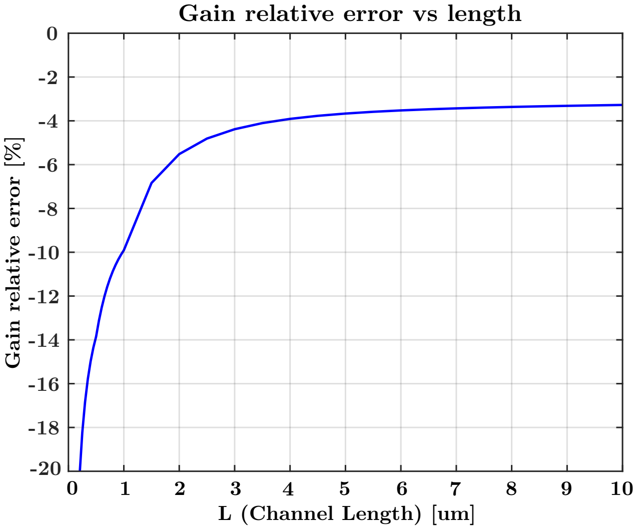


Figure 4.4: Gain error of the boxcar vs the length of the output branch transistors. This simulation uses the C_{INT} and T_{INT} values stated in Table 3.1.

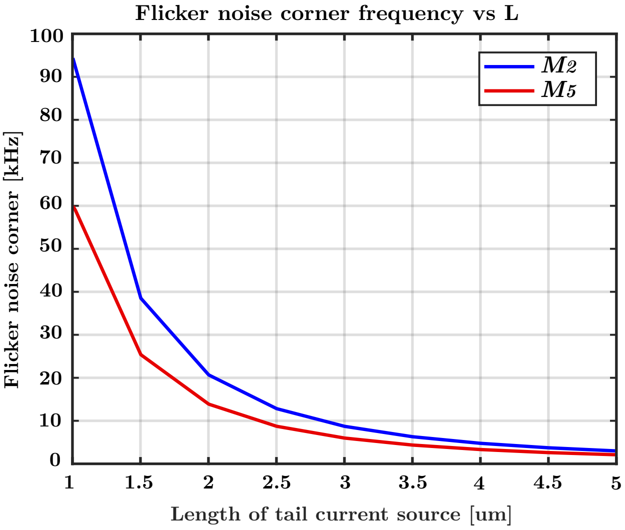


Figure 4.5: Flicker noise corner approximation of the tail transistors of the output branch vs their length.

Output and input swing

While the biasing voltages V_{bias} and V_{bn} are set to allocate the sufficient biasing currents, and V_{CM} is controlled by the common-mode feedback (CMFB) (will further be explained in subsection 4.1.4), both biasing nodes V_{bc} and V_{bp} can be set independently. These nodes influence the V_{ds} allocated for the transistors M_2 and M_5 as shown in Equation 4.11 and Equation 4.12.

$$V_{ds,M2} = V_{bc} - V_{gs,M3} \tag{4.11}$$

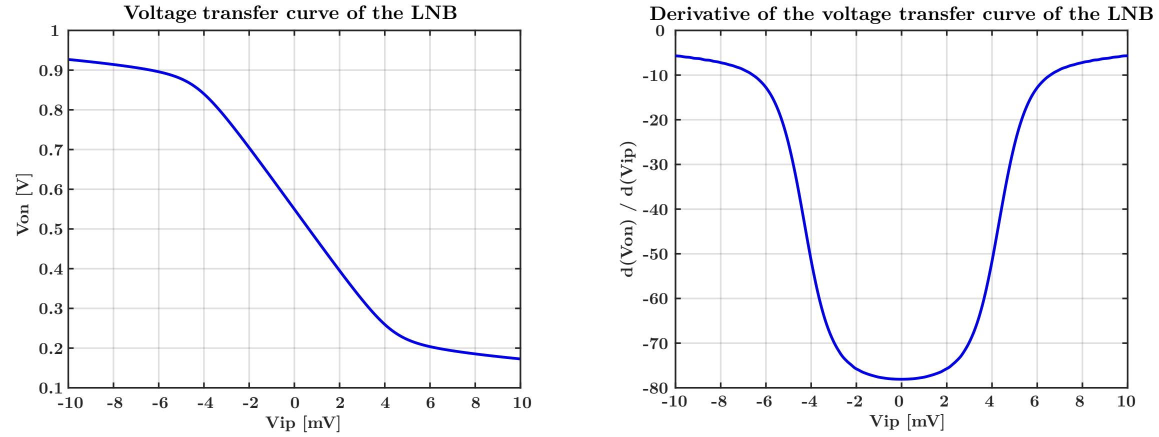
$$|V_{ds,M5}| = V_{bp} - |V_{gs,M4}| \tag{4.12}$$

To ensure sufficient headroom for M_2 and M_5 to operate in the saturation region, with their V_{dsat}

around 133 mV, V_{bc} and V_{bp} are set to achieve a V_{ds} of 200 mV. Consequently, the output swing of the amplifier is determined by the following expression (Equation 4.13).

$$V_{out,swing,pp} = V_{dd} - (V_{ds,2} + V_{ds,5} + V_{dsat,3} + V_{dsat,4}) \approx 1.1V - (0.2V + 0.2V + 0.1V + 0.1V) = 0.5V \quad (4.13)$$

Figure 4.6 illustrates the voltage transfer curve of the LNB and its derivative, simulated using the designed OTA with C_{INT} and T_{INT} values specified in Table 3.1. With the input common mode set at 0V (further explained in subsection 4.1.2), The input range of the LNB with the designed OTA is conservatively estimated at approximately 4 mVpp. This range can accommodate both the input neural signal and the residual EDO.



(a) Voltage transfer curve of the LNB.

(b) Derivative of the voltage transfer curve.

Figure 4.6: Voltage transfer characteristics of the LNB obtained with the designed OTA.

Final device sizes and parameters of the OTA

The chosen GM/ID and sizes of the transistors of the OTA are summarized in Table 4.1.

Table 4.1: Final device sizes and parameters of the OTA.

Transistor	GM/ID	W/L
M_{top}	7	1μm/4μm
M_{in}	25	26μm/1μm
M_2	15	1.2μm/4μm
M_3	20	0.6μm/4μm
M_4	20	1.6μm/4μm
M_5	15	0.6μm/4μm

4.1.2. OTA Biasing

The input DC biasing voltages and the output branch DC biasing voltages, along with the biasing currents for both the input and output branches of the OTA, are generated by a biasing circuit that is shared by all the distributed AFEs.

Input nodes biasing

The input nodes of the OTA are left floating without external DC biasing (connected to the input electrodes). This configuration is crucial for in vivo validation experiments, so it can be safely implanted in animals similar to previous μECoG recording systems [127], [128]. In these studies, the reference electrode is grounded, while the recording electrodes remain unbiased to prevent potential animal harm [128]. The use of PMOS transistors as the input pair ensures proper operation in this configuration.

Input and output branches biasing

The ASIC's power module (not designed in this work) provides a stable current (I_{bias}), this operation is modeled as an ideal current source (Fig. 4.7). This current is replicated by a current mirror to bias the input branch (in yellow). Additionally, a wide-swing cascode current mirror is employed to bias the output branch nodes (in green) [129].

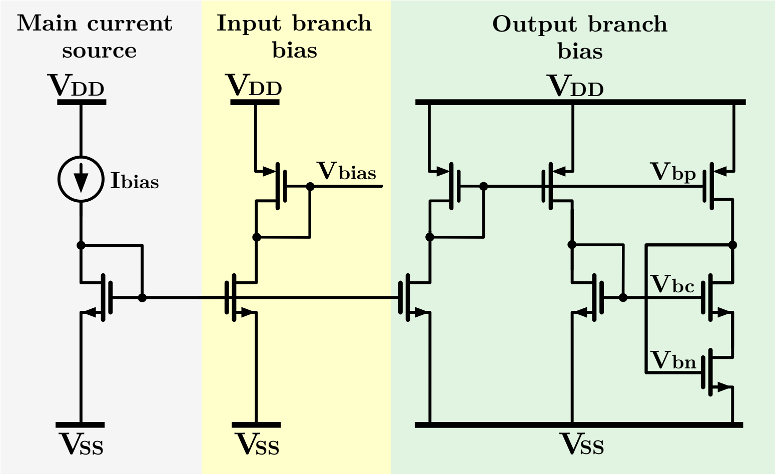


Figure 4.7: Circuit representation of the implemented bias generator.

4.1.3. Chopper stabilization

The OTA is chopped to reduce its flicker noise contribution. As previously discussed in subsection 2.2.2, the switching of the input chopper degrades the input impedance. However, as the CHS technique is employed in a DC-Coupled AFE, the degradation in input impedance is better controlled.

Similarly, if the output chopper is placed at the output of OTA, it results in a degradation of the output impedance as shown in Equation 4.14.

$$Z_{out,total} = r_{ota} \parallel \frac{1}{f_{chop} \cdot (C_{INT} + C_{par})} \tag{4.14}$$

Where C_{INT} is the integration capacitor of the boxcar sampler and C_{par} is the parasitic capacitance in the output node of the OTA. This placement of the output chopper attenuates the high output impedance of the OTA, leading to an increased gain error. To avoid this effect, the output chopper is placed at the internal nodes of the output branch of the OTA (Fig. 4.8).

The main drawback of this architecture is that the flicker noise of the cascode transistors is then not attenuated. However, their noise contribution is considered to be negligible.

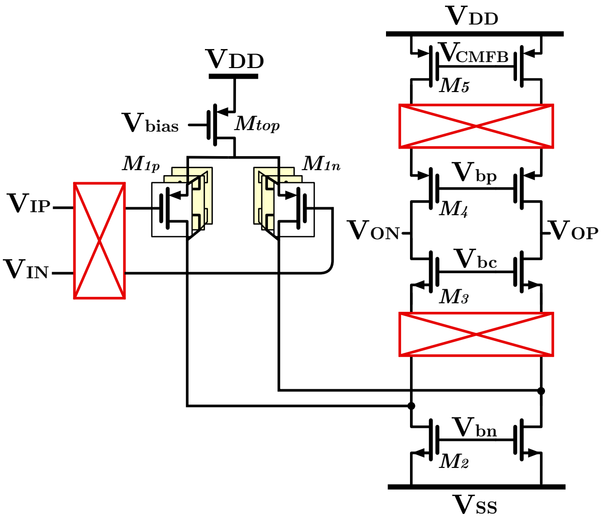


Figure 4.8: Placement of the chopping switches (highlighted in red) in the OTA circuit.

Chopping switches

A key non-ideality in chopping switches is the charge injection introduced in every switching period, as illustrated in Figure 4.9 [130].

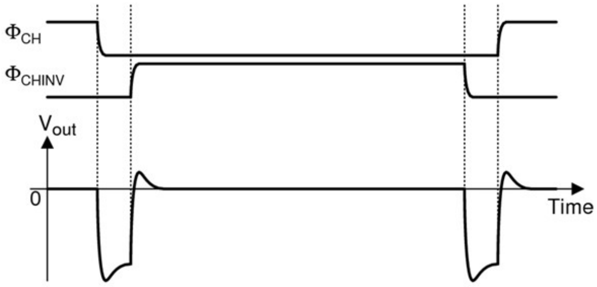


Figure 4.9: Charge injection non-ideality of chopper stabilization switches [130].

At the start of each chopping cycle, Φ_{CH} drops, injecting negative charge (for NMOS switches), followed by Φ_{CHINV} rising and injecting approximately the same charge but with positive polarity. While these charges would ideally cancel each other, various non-idealities such as switch asymmetry, parasitic capacitance differences, and clock transition variations prevent perfect cancellation. Moreover, to avoid shorting the differential pins of the chopper, a necessary ”dead zone” is introduced (further explained in section 4.3), which affects the timing of charge injection. Uncompensated charge generates a glitch in the input signal, manifesting as a high-frequency ripple at multiples of f_{chop} and induces DC offsets because the average value of the injected charge over time is not zero [130].

The chopping frequency f_{chop} is set to 50 kHz. This choice ensures alignment with the LNB sinc notches, as it is a multiple of $\frac{1}{T_{\mathrm{INT}}}$, enabling effective filtering of chopping glitches. Additionally, it exceeds the flicker noise corner of the OTA devices (Figures 4.2 and 4.5), while remaining low enough to minimize input impedance degradation. Furthermore, the system’s BLC addresses any residual DC offset resulting from chopper non-idealities.

To minimize charge injection, the size of the CMOS switches is reduced, as the injected charge is proportional to device size:

$$\Delta q = WLC_{ox}\left(V_{CLK} - V_{in} - V_{TH}\right) \tag{4.15}$$

Where Δq is the injected charge, W and L are the width and length of the CMOS switch, C_{ox} is the gate oxide capacitance per unit area, V_{CLK} is the clock voltage, V_{in} is the input voltage, and V_{TH} is the threshold voltage.

Moreover, to minimize settling error, the actual input value must be present at the AFE input for at least 90% of each chopping cycle, with a maximum of 5% allocated for tracking. The resulting static error is quantified by:

$$\varepsilon_{settling} = e^{-N} \tag{4.16}$$

Where N equals $\frac{t_{tracking}}{\tau}$, with τ determined by the $r_{on,max}$ of the sampling switch and the parasitic capacitances at the chopping switch nodes.

Input chopping switches

NMOS transistors are chosen for input switches as the input voltage is closer to V_{SS} (explained in subsection 4.1.2). With $\frac{W_n}{L_n} = 120nm/40nm$, the time constant τ=48 ns, yielding N=20.83 and negligible settling error.

The thermal noise from input chopping switches is:

$$V_{n,chop_{in}} = \sqrt{2 \cdot 4 \cdot KTR_{on} \cdot BW} = 0.394\mu V_{rms} \tag{4.17}$$

Output bottom and top chopping switches

Minimum-sized NMOS transistors are used for the bottom switches, and minimum-sized PMOS for the top switches. Their noise contributions are negligible when referred to the input.

4.1.4. Common mode feedback

Common mode feedback circuits are essential in fully differential amplifiers to ensure proper operation and stability of the amplifier. Without common-mode feedback, the common-mode voltage (the average of the positive and negative output voltages) can drift due to variations in the power supply and transistor mismatches, leading to an unreliable operation point. Common mode feedback circuits monitor the common mode voltage and adjust it to a desired level, ensuring that the amplifier operates within its optimal range, maintains high linearity, and provides accurate signal amplification [129].

In this thesis, a switched-capacitor common-mode feedback (SC-CMFB) is implemented. The main advantages of an SC-CMFB are that it imposes no restrictions on the maximum allowable differential input signals and is highly linear [129], [131].

As explained previously, the OTA is only connected to the C_{INT} during T_{INT}. Therefore, common mode stabilization is required only during that period [129]. For this reason, the simple architecture shown in Figure 4.10 is chosen.

There are two phases of operation for this circuit: the common-mode adjustment phase and the amplification phase. During the common-mode adjustment phase, the switches controlled by $\overline{\Phi_{\mathrm{INT}}}$ are closed. In this phase, the capacitors C_{cm} are charged to a value close to the given biasing voltage (V_{ocm}), which

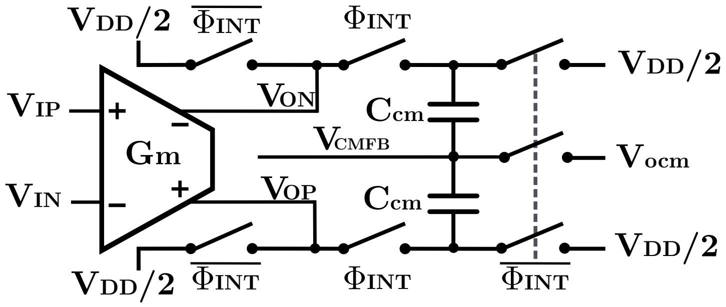

VDD/2
$\overline{\Phi\text{INT}}$
ΦINT
VDD/2
VIP
+
−
Gm
VON
Ccm
VCMFB
Vocm
VIN
−
+
VOP
Ccm
VDD/2
VDD/2
$\overline{\Phi\text{INT}}$
ΦINT
$\overline{\Phi\text{INT}}$

Figure 4.10: Circuit diagram of the implemented SC-CMFB.

is necessary to keep the common-mode output voltage at $\frac{V_{DD}}{2}$. Additionally, the output nodes are connected to $\frac{V_{DD}}{2}$ to ensure controlled operation.

During the amplification phase, the switches controlled by Φ_{INT} are closed instead. In this phase, if the common-mode output voltage is not $\frac{V_{DD}}{2}$, the generated output bias V_{CMFB} will adjust and force the common-mode output voltage back to the desired value.

It is also necessary to account for the charge transfer that occurs during the operation of the switches. Through simulation, these errors can be predicted and corrected by choosing an adequate V_{ocm}. In this implementation, a V_{ocm} of 540 mV is selected.

To choose the size of C_{cm}, the following constraints were considered. Smaller capacitive values minimize the loading effect on the output of the OTA, which would affect the gain of the boxcar. On the other hand, larger capacitive values are more robust against process variations and mismatches. Additionally, C_{cm} must be larger than the parasitic capacitances present at V_{ocm} to reduce the static error due to charge sharing. As a compromise, C_{cm} is chosen to be 50 fF (significantly smaller than C_{INT} of 4 pF).

4.1.5. Bi-level EDO compensation

The detailed implementation of the body modulation BLC, first introduced in section 3.1 is discussed in this section.

Required number of bits

To determine the number of bits required for the BLC, it is essential to ensure that the EDO compensation does not cause the signal to cross the opposite extreme threshold. For instance, if the upper threshold is exceeded, the compensation loop must be designed to prevent crossing the lower threshold as a result. As a first-order approximation, without accounting for any non-linearity of the loop, the following condition must always be satisfied:

$$IRO_{LSB} < V_{TH+} - V_{TH-} - V_{in,pp} \tag{4.18}$$

Where IRO_{LSB} is the input-referred offset generated by a least significant bit (LSB) change in the BLC, and V_{TH} is the equivalent amplitude of the threshold referred to the input range. In this example, these thresholds are set to 95% and 5 % of the input swing for the upper and lower thresholds, respectively.

The previous equation is rewritten as the following:

$$\frac{\Delta_{V_b}}{2^{N_{BLC}}} < 0.9 \cdot V_{in,swing,pp} - V_{in,pp} \tag{4.19}$$

With a voltage difference of $\Delta_{V_b} = 50$ mV (to meet the requirements specified in section 1.2), an input voltage swing of $V_{in,\text{swing,pp}} = 4$ mV, and an input signal of $V_{in,\text{pp}} = 1$mV, the minimum number of bits required is $N_{BLC} = 5$. However, due to the potential non-idealities in the implementation, an additional bit is added to enhance the robustness of the design. Therefore, the input transistors are divided into 6 parallel devices (6 bits). These transistors are sized as shown in Table 4.2 (the length of all of them is 1 μm).

Table 4.2: Dimensions of the binary coded input transistors.

Transistor	D5	D4	D3	D2	D1	D0
W [μm]	12.8	6.4	3.2	1.6	0.8	0.4

Threshold values

As mentioned earlier, the digital thresholds are implemented by digital comparators (discussed in subsection 4.4.2). The digital values of these thresholds can be freely chosen within the 2^8 codes of the D_{ADC}. However, to ensure that the ADC always operates within its linear region, the following must be considered:

The BLC has a latency in its compensation loop of 1 sample (Fig. 3.9). During this period, the D_{ADC} must not increase enough to go outside the linear region (saturate).

The EDO operates at a frequency below 0.5 Hz. Given this low frequency and the high sampling rate of $F_s = 16\,\text{kHz}$, the change in amplitude over 1 sample is negligible. However, the input neural signal causes more significant changes in amplitude. The time interval for 1 sample is $t = \frac{1}{F_s}$ seconds. The maximum rate of change for a 500 Hz signal with a peak amplitude of 0.5 mV is approximately 1570 mV/s. This results in a maximum change in signal amplitude over this interval of approximately 0.098 mV.

Converting this change to ADC codes, it is approximately 6.22 LSBs (according to the input range of the ADC, later discussed in subsection 3.2.3). Therefore, to ensure that the ADC does not saturate, the thresholds (in decimal format) should account for the following constraints:

$$TH_+ < 127 - 7 \tag{4.20}$$

$$TH_- > -127 + 7 \tag{4.21}$$

By setting these threshold values, the system ensures enough margin to react without causing the ADC to saturate, even considering the worst-case change in the input signal.

Non-linearity of the feedback loop

Until this point, the BLC is assumed to be completely linear. However, the input transistors of the OTA are biased in the subthreshold region, introducing additional non-linear factors. As explained in subsection 3.1.2, the bulk connections follow the code given by D_{BLC} (high values connect the bulk to $V_{\text{B,BLC}}$ and low values connect the bulk to V_{DD}). Considering V_g as the differential analog input at the gates of the input pair, the total output currents of each branch can be derived as shown in Equation 4.22 and Equation 4.23.

$$I_{\text{outp}} = D_{\text{BLC}} I_1 e^{\frac{V_g}{2nU_T}} + (2^N - 1 - D_{\text{BLC}}) I_0 e^{\frac{V_g}{2nU_T}} \tag{4.22}$$

$$I_{\text{outn}} = (2^N - 1) I_0 e^{-\frac{V_g}{2nU_T}} \tag{4.23}$$

where N is the number of bits used in the BLC (in this case 6), and I_1 and I_0 correspond to the drain current of a transistor when V_g is 0 and the bulk is connected to $V_{\text{B,BLC}}$ and V_{DD}, respectively. Here, n is the subthreshold slope factor and U_T is the thermal voltage. In this analysis, the source node of the input transistors is considered a virtual ground [29].

The exponential component (due to the subthreshold operation) can be expanded by a Taylor series (considering up to the third component) as shown in Equation 4.24.

$$e^{\pm\frac{V_g}{2nU_T}} \approx 1 \pm \frac{V_g}{2nU_T} + \frac{1}{2}\left(\frac{V_g}{2nU_T}\right)^2 \pm \frac{1}{6}\left(\frac{V_g}{2nU_T}\right)^3 \tag{4.24}$$

The differential output current is given by Equation 4.25.

$$I_{\text{out}} = I_{\text{outp}} - I_{\text{outn}} \tag{4.25}$$

Substituting the Taylor expansion:

$$\begin{aligned} I_{\text{out}} = &\big(D_{\text{BLC}} I_1 + (2^N - 1 - D_{\text{BLC}}) I_0\big) \left(1 + \frac{V_g}{2nU_T} + \frac{1}{2}\left(\frac{V_g}{2nU_T}\right)^2 + \frac{1}{6}\left(\frac{V_g}{2nU_T}\right)^3\right) \\ &- (2^N - 1) I_0 \left(1 - \frac{V_g}{2nU_T} + \frac{1}{2}\left(\frac{V_g}{2nU_T}\right)^2 - \frac{1}{6}\left(\frac{V_g}{2nU_T}\right)^3\right) \end{aligned} \tag{4.26}$$

After simplification, the differential output current considering binary scaling is:

$$I_{\text{out}} = D_{\text{BLC}}(I_1 - I_0) + g_m V_g + \frac{D_{\text{BLC}}(I_1 - I_0)}{2(2nU_T)^2} V_g^2 + \frac{g_m}{6(2nU_T)^2} V_g^3 \tag{4.27}$$

From the previous equation we have:

- $D_{\text{bp}}(I_1 - I_0)$: This term captures the impact of the digital back-gate input (desired input referred offset).
- $g_m V_g$: The front-gate input's main linear term which indicates that the differential output current is linearly proportional to the differential gate voltage through the input transistors g_m.
- $\frac{D_{\text{bp}}(I_1 - I_0)}{2(2nU_T)^2} V_g^2$: Highlights the interaction between the front-gate and back-gate effects, showing a second-order dependency on the front-gate voltage influenced by the digital input.
- $\frac{g_m}{6(2nU_T)^2} V_g^3$: Represents the non-linearity due to higher-order effects of the front-gate input.

The presence of these non-linear terms introduces slight distortions in the BLC's response. However, these non-linearities are not necessarily problematic for the system's overall performance. Rather, they need to be taken into consideration when designing the circuit implementation for the compensation loop.

Compensation range

While the number of bits and the sizes of the transistors in the BLC loop have already been determined, the value of ΔV_b must still be selected. As explained previously in subsection 3.1.2, the voltage applied to the bulks of the input transistors needs to be carefully controlled to prevent forward-biasing of the body diodes. Figure 4.11 illustrates the corresponding bulk diode current for the selected ΔV_b.

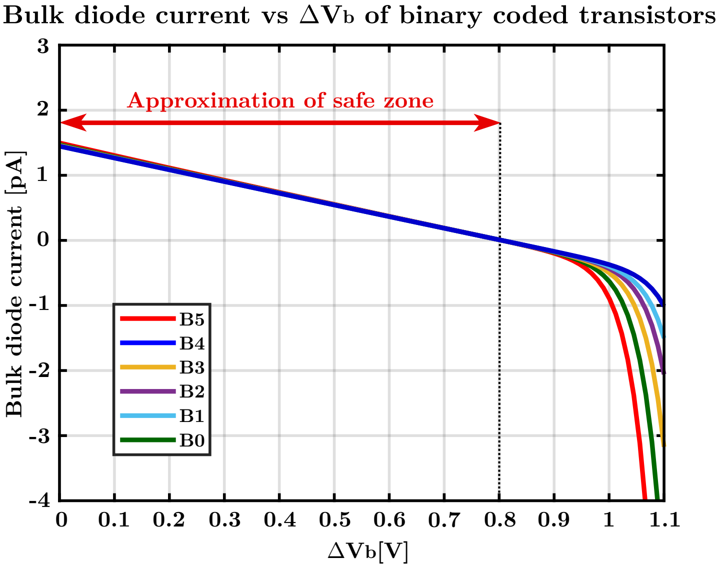


Figure 4.11: Body diode current as a function the Δ_{V_b} of binary coded PMOS devices.

ΔV_b could theoretically be chosen up to approximately 800 mV while maintaining the reverse bias condition for the body diodes. However, a lower ΔV_b must be selected in order to enhance the robustness of the design.

Due to the nonlinearity of the BLC loop, Equation 4.18 can be reformulated as:

$$\Delta_{IRO,max} < V_{TH+} - V_{TH-} - V_{in,pp} \tag{4.28}$$

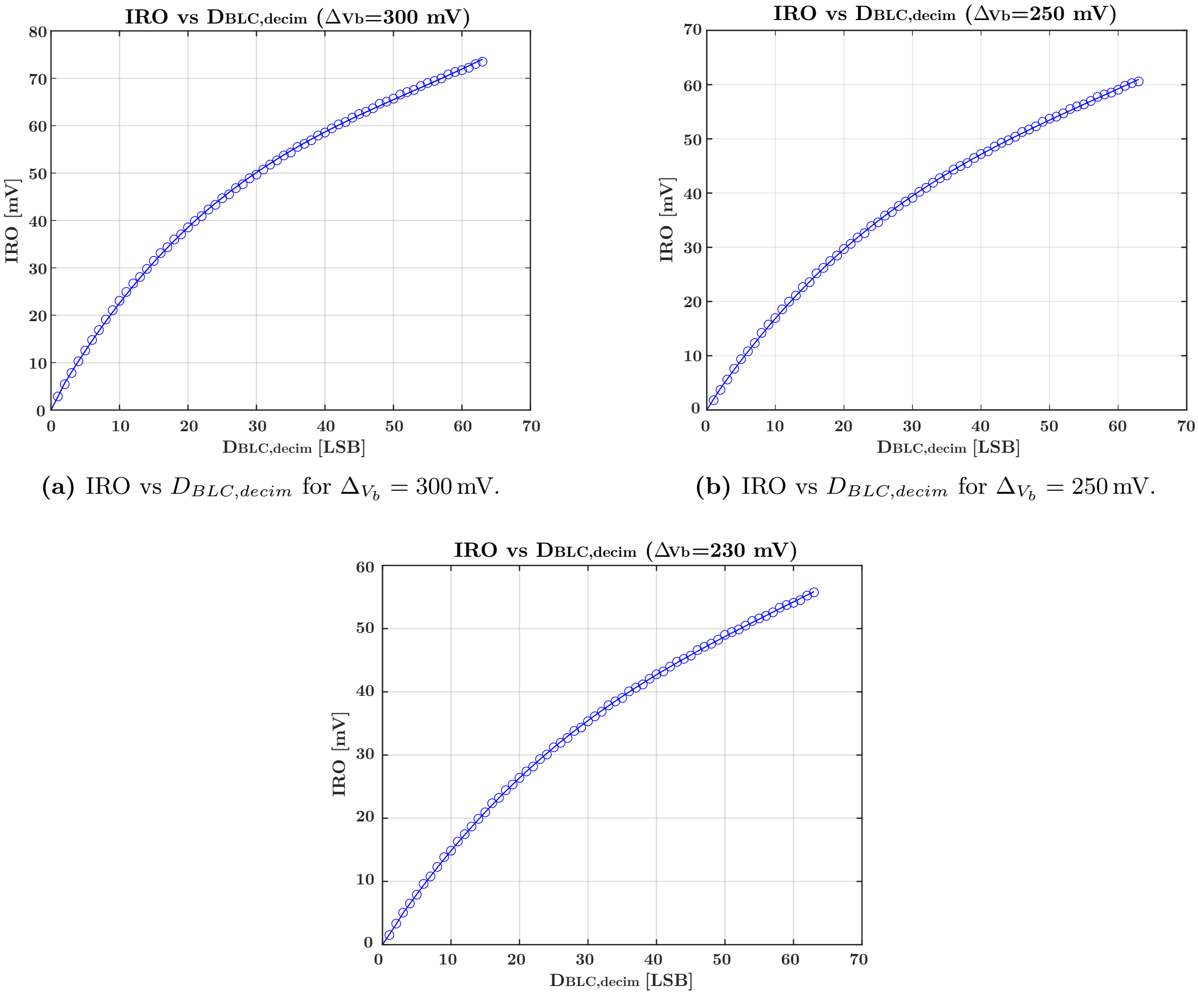
where $\Delta_{IRO,max}$ is the maximum change in the IRO with two subsequent D_{BLC} codes. Figure 4.12 illustrates the IRO generated by each code of D_{BLC} with different Δ_{V_b} values.

As expected, a higher Δ_{V_b} allows for a wider compensation range. For instance, $\Delta_{V_b} = 300\,\text{mV}$ provides a compensation range up to $\pm 74\,\text{mV}$, while $\Delta_{V_b} = 250\,\text{mV}$ provides up to $\pm 60\,\text{mV}$. However, these two choices entail a $\Delta_{IRO,max}$ of 2.6 mV and 1.8 mV respectively, which may be too large according to Equation 4.28, particularly in the presence of process variations or other non-idealities.

In contrast, $\Delta_{V_b} = 230\,\text{mV}$ results in a more manageable maximum step size of 1.5 mV and provides a compensation range up to 55 mV, which is sufficient based on the initial requirements. Therefore, $\Delta_{V_b} = 230\,\text{mV}$ is the chosen value for this design.

Chopping modulation, and polarity handling of the BLC

The offset introduced by the BLC compensation loop is chopped to counteract the up-modulated EDO. Furthermore, based on the polarity bit, D_{BLC} is assigned to control the body connections of either the positive or negative input transistor, as detailed in section 3.1 and illustrated in Figure 3.6.



(a) IRO vs $D_{BLC,decim}$ for $\Delta_{V_b} = 300\,\text{mV}$.

(b) IRO vs $D_{BLC,decim}$ for $\Delta_{V_b} = 250\,\text{mV}$.

(c) IRO vs $D_{BLC,decim}$ for $\Delta_{V_b} = 230\,\text{mV}$.

Figure 4.12: IRO vs BLC code for different Δ_{V_b} values.

Figure 4.13 shows the circuit implementation to chop the body modulation loop while still following the polarity bit.

D_{5p} to D_{0p} correspond to the body control nodes of the embedded DAC in the positive transistor, while D_{5n} to D_{0n} correspond to the negative transistor (Fig. 4.13).

The BLC chopping switches are implemented as transmission gates, incorporating both PMOS and NMOS devices. This design minimizes charge injection by enabling the opposite charge packets (holes from PMOS and electrons from NMOS) injected by the two types of devices to cancel each other out, thus reducing the overall effect of charge injection (Fig. 4.14). The total on-resistance of the transmission gate switch is given by the parallel combination of r_{onn} (NMOS on-resistance) and r_{onp} (PMOS on-resistance). These resistances vary with input voltage:

- r_{onn} is lowest when $V_{IN} = 0$
- r_{onp} is lowest when $V_{IN} = V_{DD}$

To compensate for the lower mobility of holes compared to electrons in silicon, the width of the PMOS is increased relative to the NMOS by a factor of k (Fig. 4.14) [125].

The optimal sizing ratio (k) between PMOS and NMOS is crucial to minimize charge injection cancel-

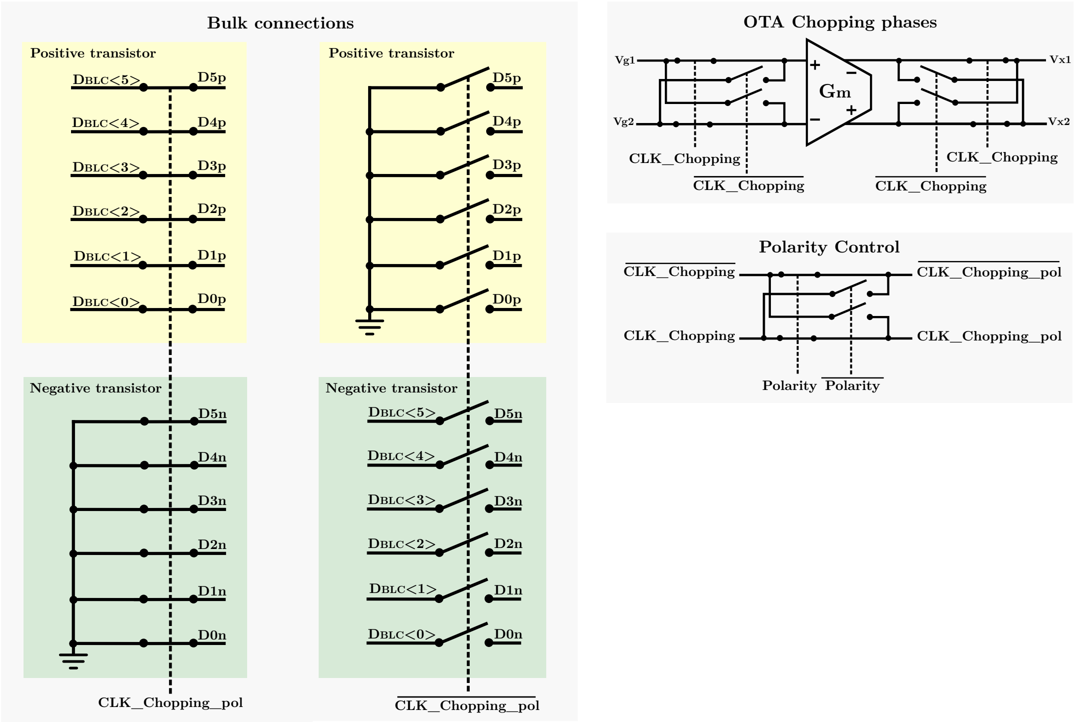


Figure 4.13: Circuit level implementation of the chopping modulation applied to the BLC loop. In this example, a positive EDO is detected (polarity=1) and it is the first phase of chopping (CLK_chopping=1).

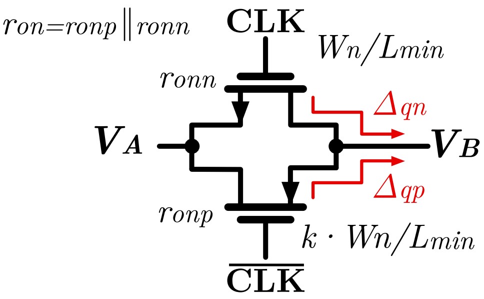


Figure 4.14: Transmission gate switch with the corresponding charge injection (in red). Minimum channel length is used, and the p-channel is sized larger than the n-channel by a factor k.

lation. The best charge injection cancellation occurs when the on-resistance (r_{on}) variation across the entire input voltage swing is minimized. Figure 4.15 illustrates this variation with different k. As shown in Figure 4.15, the optimal value of k, where the ron variation is smallest, is approximately 2.7. This point represents the best trade-off for charge injection cancellation across the input range. However, due to practical layout considerations, a ratio of 2 is chosen for this implementation.

During a chopping phase, the value must be fixed in the bulk capacitance for most of the time (> 99% of the period). Therefore, the amount of time allocated for tracking the input value corresponds to 0.5% of the chopping period (approximately 200 ns). Assuming a minimum size transmission gate with $\frac{W_p}{L_p} = \frac{240\,\text{nm}}{40\,\text{nm}}$ and $\frac{W_n}{L_n} = \frac{120\,\text{nm}}{40\,\text{nm}}$ results in a negligible settling error.

Moreover, the input-referred noise of these switches is derived as shown in Equation 4.29.

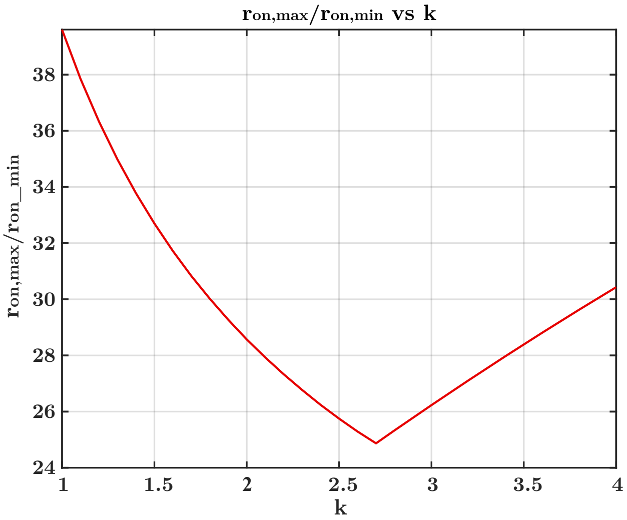


Figure 4.15: On-resistance variation as a function of p-channel/n-channel sizing ratio (k).

$$V_{n,dsl_{switch_in}} = \sqrt{12 \cdot 4 \cdot KTR_{on} \cdot \left(\frac{gmb}{gm}\right)^2 \cdot \mathrm{BW}} = 135\mathrm{nV_{rms}} \tag{4.29}$$

Where $\frac{gmb}{gm}$ is approximated to be 1/7 in this present technology.

4.1.6. Sampling and SC-LPF

The boxcar sampler is composed of three switches corresponding to (T_{INT},T_{LPF} and T_{RST}) (Fig. 3.1). These switches require careful design regarding charge injection, settling errors, and leakage. To mitigate charge injection effects, a fully differential sampling approach is employed, making charge injection a common-mode phenomenon to a first order.

As these three switches (T_{INT}, T_{LPF}, and T_{RST}) require operation across a wide input range or at precisely $\frac{V_{DD}}{2}$, single PMOS or NMOS switches are insufficient. To overcome this limitation, these switches are implemented as transmission gates (Fig. 4.14), which accommodate a broader range of input voltages.

Integration switch

For the T_{int} controlling switch, minimum lengths and widths are used to minimize charge injection. The used sizes for these switches are $\frac{W_p}{L_p} = \frac{240nm}{120nm}$ and $\frac{W_n}{L_n} = \frac{120nm}{40nm}$.

Reset switch

In the reset and LPF switches, the static error can be determined as previously stated in Equation 4.16. As explained in chapter 3, the switch reset is used to reset both positive and negative nodes of the boxcar sampler to the common mode voltage ($\frac{V_{DD}}{2}$). Since this switch is turned off approximately 98% of the sampling period, sub-threshold leakage is a concern. The amplitude of the channel sub-threshold leakage can be derived as shown in Equation 4.30 [132].

$$I_{\mathrm{leak}} = I_{\mathrm{const}}\frac{W}{L} \cdot e^{(V_{\mathrm{gs}}-V_{\mathrm{th}})/nU_T} \cdot \left(1 - e^{-V_{\mathrm{ds}}/U_T}\right) \cdot \left(e^{\eta V_{\mathrm{ds}}/nU_T}\right) \tag{4.30}$$

Where I_{const} is a technology-dependent constant, n is the subthreshold slope factor and U_T is the thermal voltage. The resulting error introduced by leakage can be derived as in Equation 4.31.

$$\Delta V_{leak,C_{INT}} = \frac{I_{leak} \cdot (T_s - T_{RST})}{C_{int}} \tag{4.31}$$

To minimize this unwanted error, thick oxide devices are used to increase their corresponding threshold voltage, and their lengths are increased. Therefore, the sizes used are $\frac{W_p}{L_p} = \frac{1.28\,\mu m}{320\,nm}$ and $\frac{W_n}{L_n} = \frac{640\,nm}{320\,nm}$. The corresponding leakage-induced error is below $80\,\mu V$ (considered acceptable when referred to the input). Furthermore, $N = 16.48$, leading to a negligible settling error.

Low pass filter switch

As discussed previously in chapter 3, the low-pass filter switch is responsible for sampling the previously integrated voltage in C_{int} to the second capacitor C_{LPF}. This sampling circuit faces four main challenges: charge injection, settling error, leakage, and hold-mode feedthrough through the C_{ds} of the switch. Feedthrough occurs when the AC signal unintentionally couples through the switch's drain-source capacitance (C_{ds}) while the switch is in the OFF state, potentially distorting the held voltage. A T-Switch scheme is implemented to mitigate this AC signal coupling, as shown in Figure 4.16 [132].

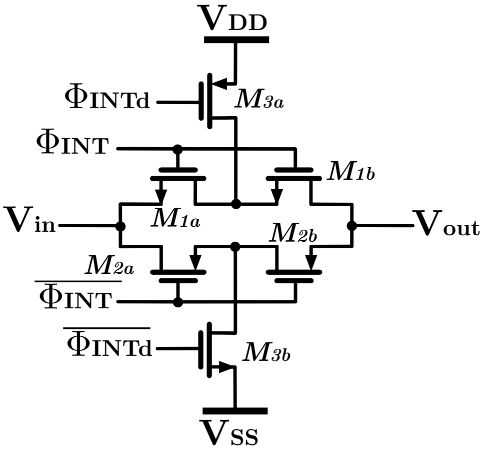


Figure 4.16: Analog T-switch implemented for the low-pass filter switch.

The T-switch configuration effectively addresses the feedthrough problem by introducing a third transistor (M3) between the two main switching transistors (M1 and M2). When the switch is in the OFF state, M3 creates a high-impedance node between M1a and M1b (also M2a and M2b). This high-impedance node significantly attenuates any AC signal that might couple through the C_{ds} of M1a (M2a). The residual coupled signal is further attenuated by the C_{ds} of M1b (M2b), resulting in greatly reduced feedthrough to the output.

This scheme utilizes two delay signals ($\overline{\Phi_{\text{INTd}}}$ and Φ_{INTd}), generated by the non-overlapping clock generator (further explained in section 4.3). The transistor sizes are $\frac{W_p}{L_p} = \frac{240nm}{40nm}$ and $\frac{W_n}{L_n} = \frac{120nm}{40nm}$. With these sizes, a τ of 29.34 ns is obtained, resulting in N=21.56 and a minimal settling error.

The corresponding leakage-induced error due to this switch is determined by:

$$\Delta V_{leak,C_{LPF}} = \frac{I \cdot (T_s - T_{LPF})}{C_{int}} \tag{4.32}$$

The leakage-induced error due to this circuit is below 1 μV, which is considered negligible when referred to the input.

4.2. SS ADC

As already mentioned, the main core of the ADC is composed by global ramp generator and a comparator that behave as a voltage-to-time converter (VTC), and a 8-bit time-to-digital converter (TDC) implemented as a counter (discussed in detail in subsection 4.4.1). The global ramp generator circuit could be implemented following the approach described in [51]. However, in this thesis, the implementation of this circuit has not been carried out. Instead, an ideal voltage source is utilized to model the ramp generator in the simulations conducted.

The ADC's reference voltage (V_{REF}) determines the ramp signal range and is based on the boxcar sampler's output swing, as discussed in Equation 4.13. With the boxcar sampler providing a gain of approximately 150 V/V and the OTA input swing being about 4 mV_{pp}, the amplified output swing reaches approximately 600 mV_{pp}. To accommodate potential variations in input signal and gain, a safety margin is incorporated by setting the ADC's V_{REF} to 320 mV, corresponding to a full-scale range of 640 mV$_{pp}$ at the ADC input. This arrangement ensures adequate headroom for the amplified signal. The resulting input range of the ADC can be expressed as:

$$\mathrm{V_{CM} - V_{REF} \leq ADC_{IN} \leq V_{CM} + V_{REF}} \tag{4.33}$$

Where V_{CM} is the common mode imposed by the boxcar sampler and is equal to $\frac{V_{DD}}{2}$.

4.2.1. Continuous time comparator

Continuous and discrete time (dynamic) comparators can be part of a SS-ADC. Continuous-time comparators (CTC) consist of a MOS differential pair, biased with a current source and followed by an additional gain stage, continuously monitoring the input voltage. On the other hand, dynamic comparators [133], often implemented as regenerative latches, perform comparisons at regular intervals, dictated by a clock signal. The continuous operation of CTCs results in drawing a mean I_{bias} current during $T_{conversion}$, dominating the power consumption of the ADC. Dynamic comparators, on the other hand, only draw power when a comparison is effectuated. However, as 2^8 comparisons are required for an 8-bit SS-ADC, this approach can entail a similar power consumption to a CTC implementation. Moreover, while both types of comparators can introduce kickback noise (an unwanted transient noise generated during switching), dynamic comparators typically exhibit more severe kickback effects due to their high frequency of switching events [133].

Therefore, to compare the distributed ramp signal (reference) with the sampled signal, a continuous-time comparator similar to the one employed in [51] is used. The implemented architecture for the comparator consists of two stages: the first being a differential stage with active load (with 4 inputs to accommodate a differential input signal and a differential ramp), followed by a common-source gain stage (Fig. 4.17). In addition, an output inverter is used to provide a sharper transition.

The comparator must make a decision during the interval comprised in $T_{LSB} = \frac{T_{conversion}}{2^8/2}$. Therefore, the minimum bandwidth required for the comparator is defined in the following equation:

$$BW \geq \frac{N}{2\pi T_{LSB}} \tag{4.34}$$

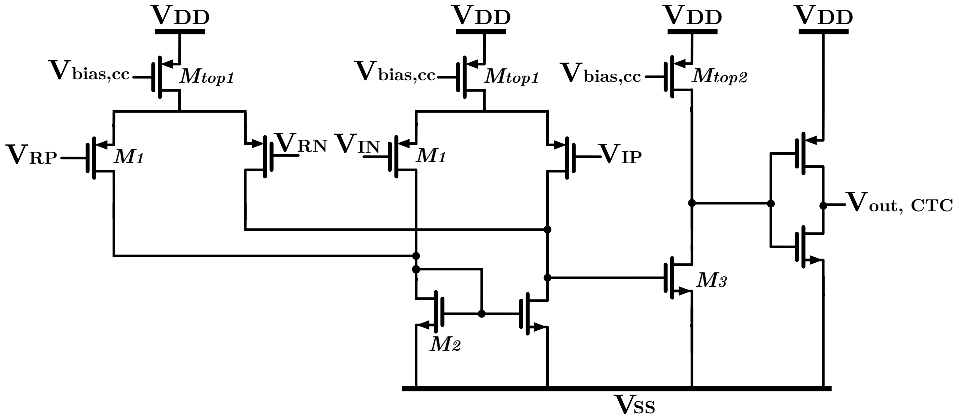


Figure 4.17: Schematic representation of the comparator implemented in the SS ADC.

N determines the bandwidth redundancy. In this design, the parameter N is chosen to be 4 to increase robustness. The bandwidth of the CTC will vary depending on the DC common mode present in the inputs when making a decision. Therefore, the requirement set in Equation 4.34 must be met across the entire common mode input operation range. The two main poles that influence this comparator are derived in Equation 4.35 and Equation 4.36.

$$p_1 = \frac{1}{C_{L1} \cdot r_{out1}} \tag{4.35}$$

where $C_{L1} = C_{gg,M3} + 2 \cdot C_{dd,M1} + C_{dd,M2}$ and $r_{out1} = \frac{r_{o,M1}}{2} \parallel r_{o,M2}$

$$p_2 = \frac{1}{C_{L2} \cdot r_{out2}} \tag{4.36}$$

where $C_{L2} = C_{in,Inv} + C_{dd,M3} + C_{dd,Mtop2}$ and $r_{out2} = r_{o,Mtop2} \parallel r_{o,M3}$

The small-signal gain of the first stage and overall gain of this two-stage comparator are derived in Equation 4.37 and Equation 4.38.

$$A_{v1,Comp} = g_{m,M1} \cdot (r_{o,M2} \parallel r_{o,M1}) \tag{4.37}$$

$$A_{v,Comp} = A_{v1,Comp} \cdot g_m(r_{o,Mtop2} \parallel r_{o,M3}) \tag{4.38}$$

Therefore, the frequency response of the two-stage comparator can be expressed according to Equation 4.39.

$$A_v(s) = \frac{A_{v,Comp}}{\left(\frac{s}{p_1} + 1\right)\left(\frac{s}{p_2} + 1\right)} \tag{4.39}$$

Another important factor to consider is the offset, which affects the precise point where the comparator output changes from high to low. Although this offset, caused by mismatch or process variations, is also linearized by the BLC loop, the systematic offset of the circuit can be minimized through careful design of the comparator. This minimization helps ensure the correct functionality of the overall system.

To minimize this offset, the following equalities must be considered:

$$V_{GS,M3} = V_{GS,M2} = V_{DS,M2} \tag{4.40}$$

$$\frac{I_{DS,Mtop1}}{I_{DS,Mtop2}} = \frac{I_{DS,M2}}{I_{DS,M3}} \tag{4.41}$$

Lastly, the input-referred noise of the comparator cannot be derived analytically because it produces a digital output. Therefore, it must be obtained through simulations and probabilistic studies [134]. However, it is possible to identify the main noise contribution factors. Equation 4.42 exposes the relation within the devices g_m and their input-referred thermal noise contribution.

$$\overline{V_{n,int}^2} \propto \left(\frac{1}{g_{m,M1}} + \frac{g_{m,M2}}{g_{m,M1}^2}\right) + \left(\frac{1}{A_{v1,Comp}^2} \cdot \left(\frac{1}{g_{m,M3}} + \frac{g_{m,Mtop2}}{g_{m,M3}^2}\right)\right) \tag{4.42}$$

From the previous derivation, it is possible to conclude that to minimize the noise of the comparator, M_1 must be biased in a large G_M/I_D ratio (weak inversion). At the same time, M_2 (and consequently also M_3) must entail a lower G_M/I_D ratio (moderate inversion). In addition, M_{top} devices must have an even lower G_M/I_D ratio (strong inversion) to reduce their noise contribution, which is also beneficial for the current source matching.

Final device sizes and parameters

Considering the aforementioned considerations, a biasing current of 400 nA is chosen for $I_{DS,M_{\text{top1}}}$ and $I_{DS,M_{\text{top2}}}$. To fulfill Equation 4.40 and Equation 4.41, the same $\frac{G_m}{I_D}$ ratio is assigned to both M_{top1} and M_{top2}, resulting in equal sizing and V_{GS} for these transistors. Similarly, M_2 and M_3 are designed with identical $\frac{G_m}{I_D}$ ratios, which leads to the same sizes and V_{GS} for these transistors as well.

The final device sizes and their corresponding G_M/I_D ratios (when the input common-mode is equal to V_{CM}) are presented in Table 4.3.

Table 4.3: Final device sizes and parameters of the CTC.

Transistor	G_M/I_D	W/L
M_{top1}	10	200 nm/900 nm
M_{top2}	10	200 nm/900 nm
M_1	23	700 nm/300 nm
M_2	16	200 nm/900 nm
M_3	16	200 nm/900 nm

4.3. Clock generation

Typically, to generate the clock lines controlling the boxcar sampler (T_{INT}, T_{RST}, T_{LPF}), chopping, and the digital logic, an on-chip clock generator would be used. As implemented in [51], all these signals can be derived from the high-frequency clock used for the SS ADC counter (CLK) using delay lines. However, in this thesis, these clock signals are instead generated using Verilog-A for simulation purposes (Appendix B).

The main clock frequency is selected to be 4.24 MHz, which corresponds to 265 clock periods per sampling cycle (eq. 4.43).

$$N_{clk} = \frac{f_{main}}{f_s} = \frac{4.24\text{ MHz}}{16\text{ kHz}} = 265 \tag{4.43}$$

where N_{clk} is the number of clock periods per sampling cycle, f_{main} is the main clock frequency, and f_s is the sampling frequency. The 265 clock periods are allocated as illustrated in Figure 4.18.

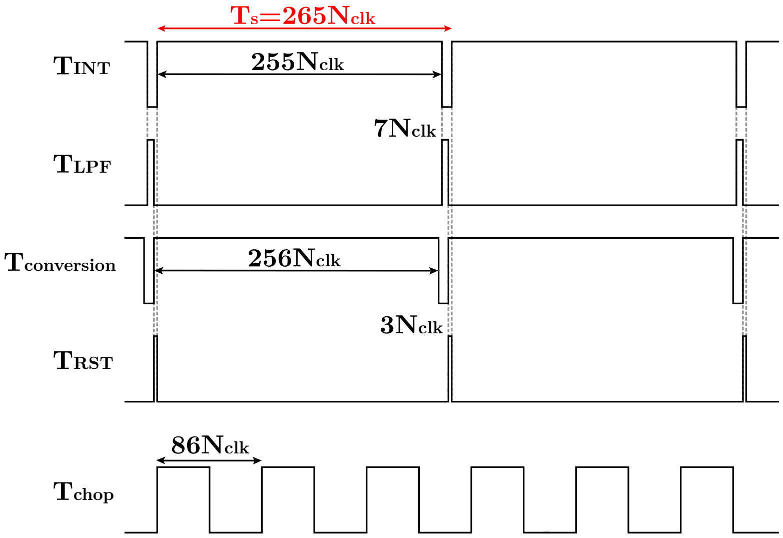


Figure 4.18: Signals generated by the on-chip clock generator and the corresponding N_{clk} allocated.

For the implementation of the chopping blocks (subsection 4.1.3) or the T-switch (subsection 4.1.6), which require non-overlapping signals, a non-overlapping clock generator is required. Figure 4.19 illustrates the circuit implemented to generate the non-overlapping signals.

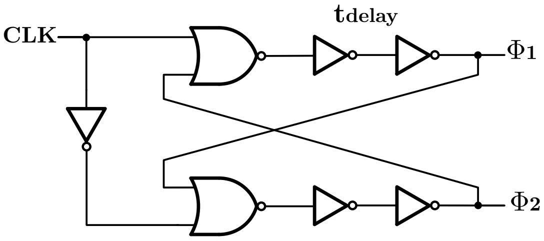


Figure 4.19: Circuit design of the non-overlapping clock generator.

As shown in the previous figure, two non-overlapping signals, Φ_1 and Φ_2, are generated from a given clock signal with a desired delay. These signals are designed to ensure that they do not overlap, maintaining the necessary timing separation for proper operation of the circuit.

4.4. Digital logic

The previously mentioned ADC counter and the bulk modulation compensation loop logic are implemented with high-threshold voltage (hvt) cells from the TSMC PDK library to reduce their static power consumption. The system includes an initialization procedure at startup. When the operation begins, all sequential logic blocks are reset to a known state using their respective clear signals.

4.4.1. ADC digital logic

The ADC logic consists of two main components: the counter required for the SS-ADC and the logic for the accumulation procedure that results from the oversampling operation.

SS ADC Counter

A TDC implemented as a counter is utilized to achieve a digital output of the SS-ADC. This implementation is shown in figure Figure 4.20. The counter is constructed using 8 D-Flip Flops (DFF) connected asynchronously (seen in yellow). The output digital word of this counter ($D_{ADC} < 7:0 >$) increments by one code at every clock cycle when both $V_{out,CTC}$ and Not_sat signals are high. To prevent the ADC from overflowing, additional logic (shown in gray) is implemented. This logic generates the Not_sat signal, which remains high until the counter reaches its maximum value of ”11111111”. Once this maximum is reached, Not_sat goes low, effectively stopping the counter from incrementing further. After every ADC conversion, the ADC output $D_{ADC} < 7:0 >$ is evaluated by the BLC logic (see subsection 4.4.2) and accumulated by the oversampling accumulation logic. After that, the counter is reset before the next ADC conversion (by a low voltage on CL_{ADC}).

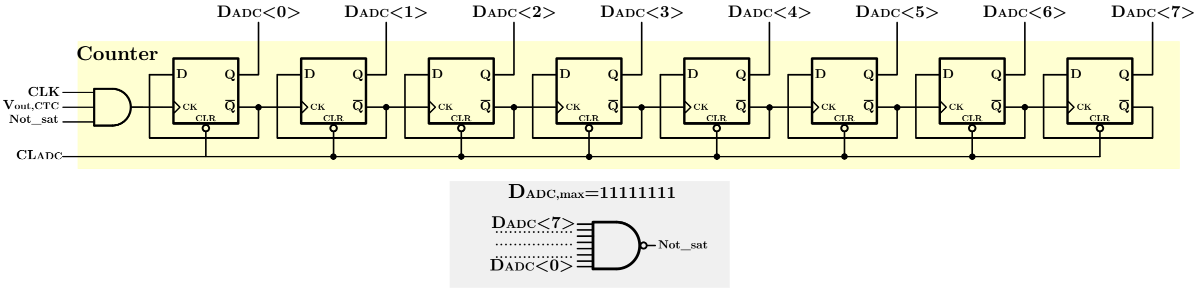


Figure 4.20: Implementation of the 8-bit counter used in the SS-ADC (yellow) and anti-overflowing logic (grey).

Oversampling accumulation logic

Figure 4.21 illustrates the logic implementation to perform the oversampling and accumulation operation explained in subsection 3.2.3.

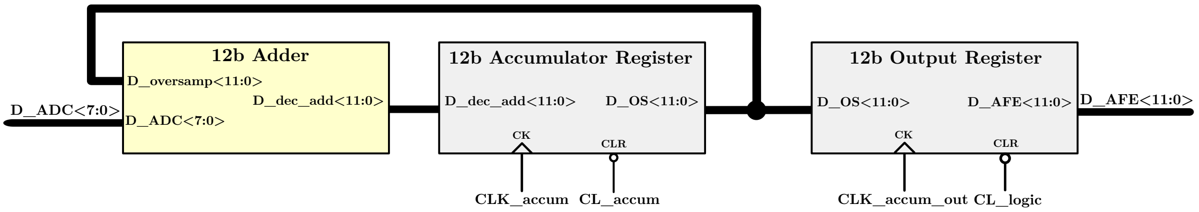


Figure 4.21: Schematic of the logic used for the accumulation process due to oversampling.

The 8-bit output from the SS-counter ($D_{ADC} < 7:0 >$) is accumulated over 16 samples and stored in the 12-bit Accumulator Register, which is built using 12 DFFs. This addition is carried out by the 12-bit Adder (highlighted in yellow). Once this accumulation process is completed, the result (D_{OS}) is stored into the 12-bit Output Register, also constructed with 12 DFFs. This register holds the accumulated value as the final output of the AFE (D_{AFE}) for the following 16 additional operations. Afterwards, the 12-bit Accumulator Register is reset and the cycle is repeated.

The implementation of the 12-bit Adder is shown in Figure 4.22.

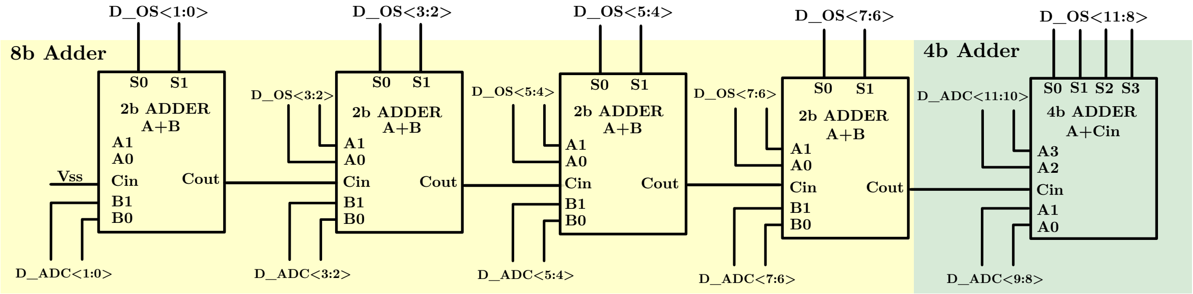


Figure 4.22: Schematic of 12-bit adder.

To reduce the design complexity, the initial bits of the addition operation between the already accumulated value ($D_{OS} < 11 : 0 >$) and the ADC output ($D_{ADC} < 7 : 0 >$) are performed using four 2-bit adders (shown in yellow). Additionally, the four most significant bits (MSBs) of the summation operation are implemented with a 4-bit adder (illustrated in green). This 4-bit adder sums the four MSBs of $D_{OS} < 11 : 0 >$ along with the carry from the previous 8-bit addition operation. The circuit design and the corresponding logic table of these blocks are shown in Appendix C.

4.4.2. BLC digital logic

As previously explained in section 3.1, the logic corresponding to the BLC compensation loop contains two digital comparators to set the upper and lower thresholds and the digital logic to increase or decrease the compensation code ($D_{BLC} < 5 : 0 >$) stored in a register (Fig. 4.23).

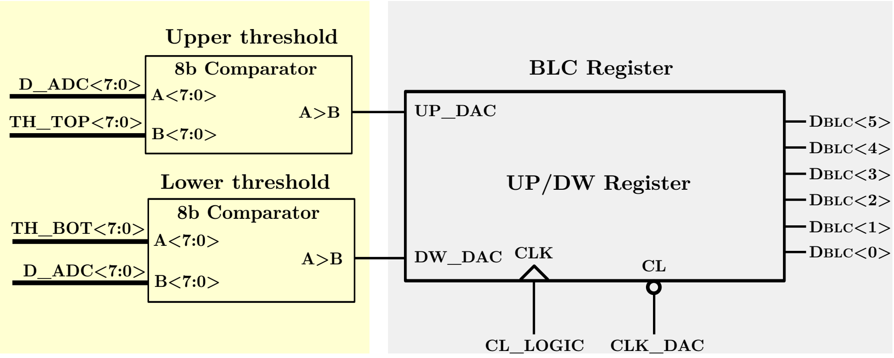


Figure 4.23: Schematic of the logic used for the BLC.

The D_{BLC} code may increase only up to ”111111” and decrease only to ”000000”, and must not overflow in any case. In addition, the logic must keep track of the polarity of the offset to choose the modulated transistor (see Fig. 4.13). The flowchart of this control loop is illustrated in Figure 4.24.

In the initial operation, the $D_{BLC,decim}$ starts at 0, and the polarity bit (Pol) is arbitrarily set to 0 (representing a negative offset). The operation of this logic can be understood through two main scenarios:

- **Positive Offset (Pol = 1):**
  - If the upper threshold (TH+) is crossed (indicating the amplitude of the positive EDO is increasing), D_{BLC} is incremented by one code. If D_{BLC} has already reached its maximum value, a flag (STOP_UP) is activated, preventing any further increase.
  - If the lower threshold (TH-) is crossed (indicating the amplitude of the positive EDO is decreasing), D_{BLC} is decremented by one code. If D_{BLC} is already at its minimum value,

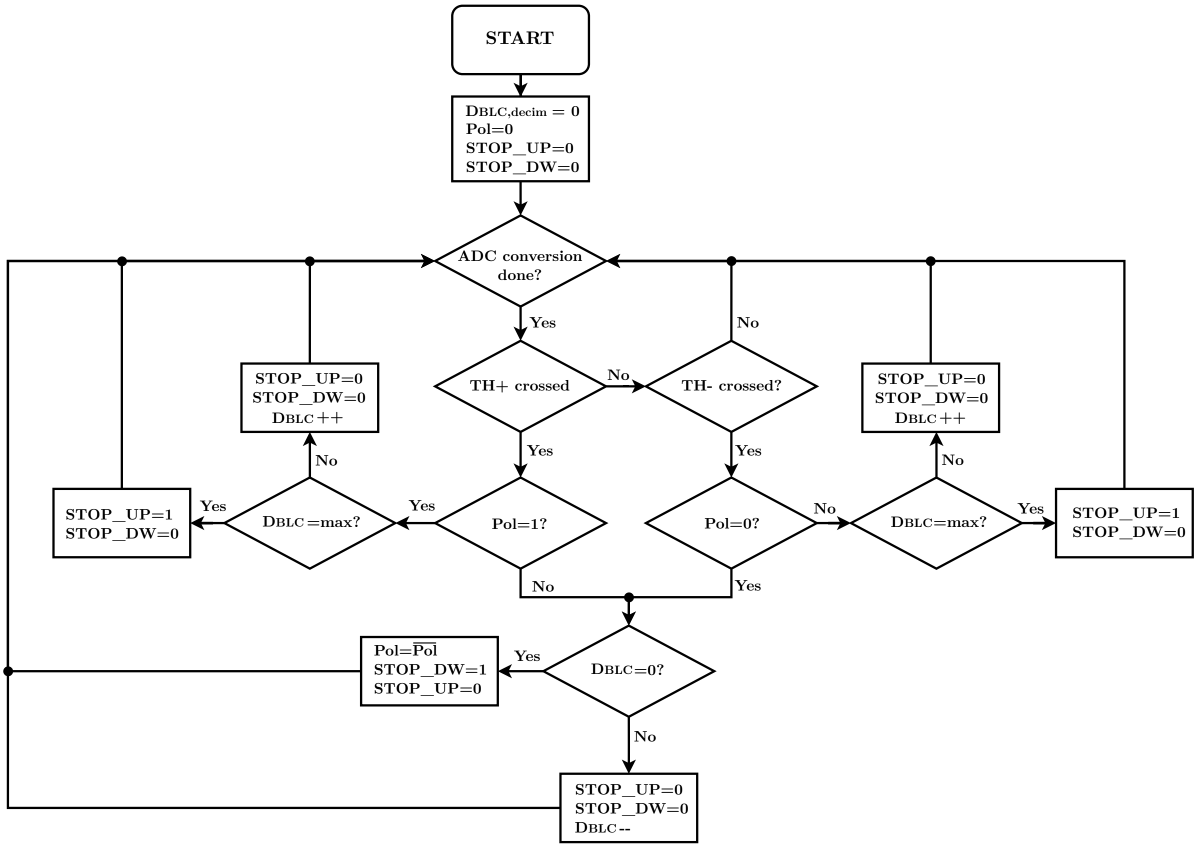


Figure 4.24: Flowchart of the BLC control loop.

a flag (STOP_DW) is activated, further decrease is halted, and the polarity (Pol) bit is toggled (signaling a transition from a positive EDO to a negative EDO).

- **Negative Offset (Pol = 0):**
  - If the upper threshold (TH+) is crossed (indicating the EDO is decreasing), D_{BLC} is decremented by one code. If D_{BLC} has already reached its minimum value, a flag (STOP_DW) is activated, preventing any further decrease and the polarity (Pol) bit is toggled (signaling a transition from a negative EDO to a positive EDO).
  - If the lower threshold (TH-) is crossed (indicating the EDO is increasing), D_{BLC} is incremented by one code. If D_{BLC} is already at its maximum value, a flag (STOP_UP) is activated and further increase is halted.

UP-DOWN Register

An UP/DW counter is implemented with 6 T-FFs (Fig. 4.25), along with the corresponding logic using AND and OR gates to decide whether to increase or decrease a code (highlighted in yellow). Additionally, it includes the logic control combinational block (shown in gray), which updates the polarity bit and generates the previously mentioned STOP flags. The polarity control (illustrated in green) adjusts according to the polarity whether to increase (UP) or decrease (DW) the code when necessary.

This functionality of the entire UP-DOWN register and the DAC control logic are summarized in Table 4.4 and Table 4.5 respectively.

Where UP_DAC and DW_DAC are the output signals from the digital comparators, and Current Pol is the current polarity bit (before the following evaluation). The gate-level circuit design of this block along with the corresponding complete truth table is displayed in Appendix C.

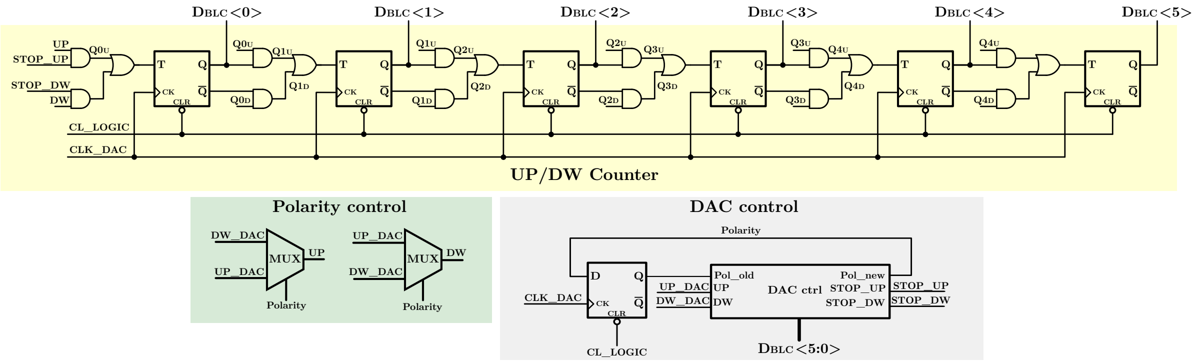


Figure 4.25: Implementation of the 6-bit DAC logic.

Table 4.4: Operation principle of the 6-bit UP-DOWN register.

Inputs			**Outputs**	
Cur Pol	**UP**	**DW**	**Next D_{BLC}**	**Next Pol**
0	1	0	$D_{BLC} - 1$ unless min	1 if min else 0
0	0	1	$D_{BLC} + 1$ unless max	0
1	1	0	$D_{BLC} + 1$ unless max	1
1	0	1	$D_{BLC} - 1$ unless min	1 if min else 0
Any	0	0	D_{BLC}	Same as initial
Any	1	1	D_{BLC}	Same as initial

Table 4.5: Operation principle of the DAC control combinational logic, showing only the cases where the outputs change.

Inputs				**Outputs**		
Current Pol	D_{BLC}	**UP ADC**	**DW ADC**	**STOP UP**	**STOP DW**	**Polarity**
0	000000	1	0	0	1	1
0	000000	0	1	0	0	0
1	000000	1	0	0	0	1
1	000000	0	1	0	1	0
0	111111	1	0	0	0	0
0	111111	0	1	1	0	0
1	111111	1	0	1	0	1
1	111111	0	1	0	0	1

5

Simulation results

This chapter presents simulation results used to verify the functionality and performance of the circuit and to ascertain whether the system meets the requirements presented in Table 1.1. These simulations were conducted with the system implemented according to the design principles outlined in chapter 3 and chapter 4.

5.1. Transfer function

Figure 5.1 illustrates the transfer function resulting from the LNB and the SC-LPF previously discussed in chapter 4.

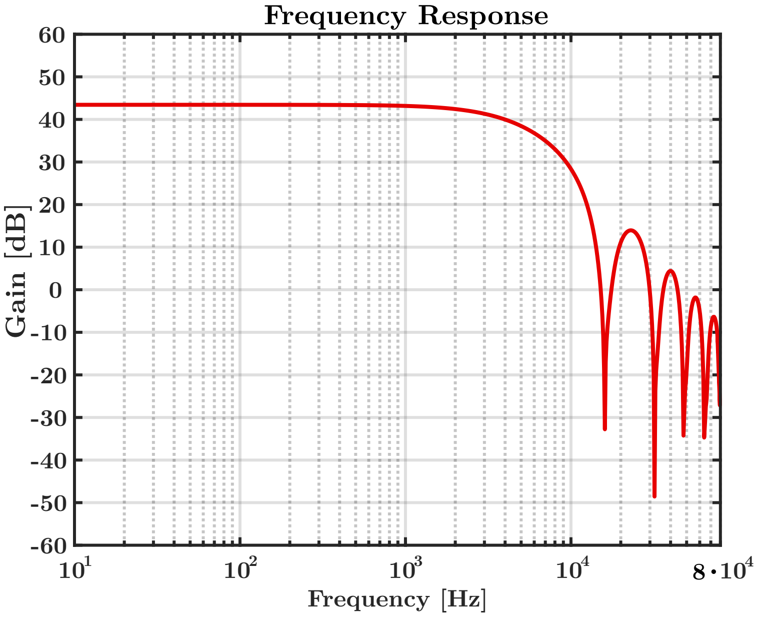
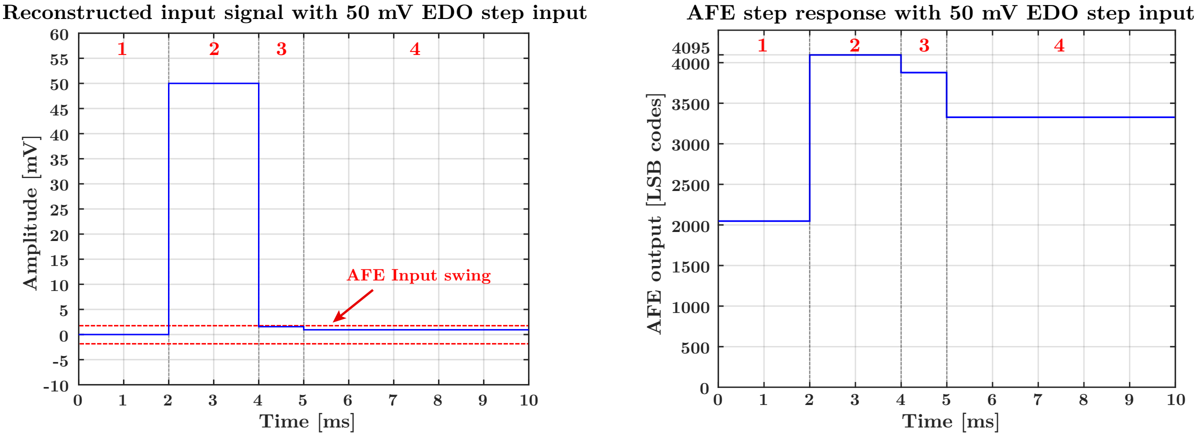


Figure 5.1: Frequency response of the LNB and SC-LPF, showing a DC gain of 43.4 dB.

The AFE response before digital accumulation shows a 43.4 dB mid-band gain and a -40 dB/dec roll-off (from the LNB's sinc function and the SC-LPF's response), resulting in a bandwidth of 3.61 kHz (which is later reduced by the digital accumulator) and notches at multiples of F_S, as expected.

5.2. System step response

To verify the initial operation of the designed AFE, a 50 mV unit-step signal is applied to the positive input, simulating the EDO (Fig. 5.2).



(a) Reconstructed input signal.

(b) AFE output.

Figure 5.2: AFE step response to a 50 mV unit-step input signal simulating EDO, with different operating regions highlighted in red. The BLC upper threshold (TH+) is set to 11111111, equivalent to 255 LSB codes.

Figure 5.2 illustrates the system's response to a step signal applied at 2 ms. This input immediately saturates the AFE as it far exceeds the AFE's input swing (region 2). The BLC initiates compensation for the EDO, and D_{BLC} increases code by code until the AFE is no longer saturated. In region 3, the AFE transitions into its linear region. After 5 ms (region 4), the output (in the linear region) is static, with a small residual offset of 1.33 mV when referred to the input. The output voltage difference between regions 3 and 4 arises from the D_{ADC} output being averaged during accumulation. While compensation is complete in region 3, the accumulated output (AFE output) still incorporates some saturated samples, resulting in a slightly higher average value compared to region 4.

5.3. Noise analysis

Figure 5.3 illustrates the Input-Referred Noise (IRN) Power Spectral Density (PSD) with and without chopping modulation at a frequency of 50 kHz.

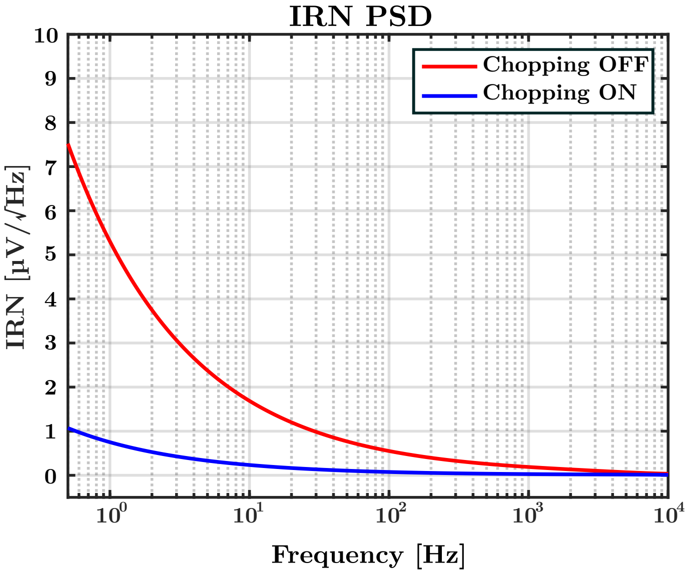


Figure 5.3: Simulated IRN PSD with chopping OFF and with chopping ON.

When chopping modulation is activated, the flicker noise contribution is significantly reduced, from an IRN of 14.48 μV_{rms} to a lower IRN of 1.69 μV_{rms} between 0.5 Hz and 500 Hz. Additionally, the input impedance of the Analog Front End (AFE) with this chopping frequency is 71.38 MΩ. A higher chopping frequency would be required to reduce further or completely eliminate the flicker noise, albeit with diminishing returns in noise performance.

5.4. Frequency Spectrum

Figure 5.4 displays the output frequency spectrum with transient noise when a $1mVpp$ sine signal with a frequency of 100 Hz is applied to the input. The signal Integrity and performance metrics (with and without noise) are displayed in Table 5.1.

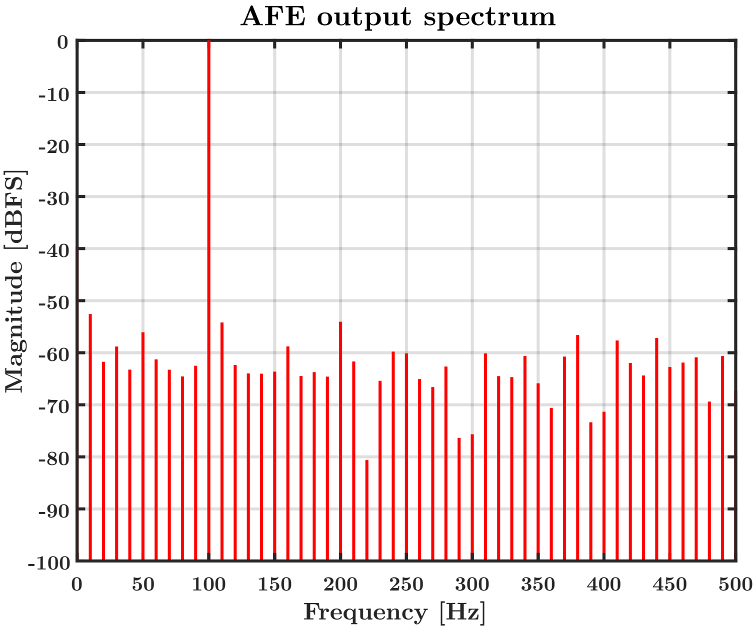


Figure 5.4: Frequency spectrum of a 1 mV_{pp} sinewave with a frequency of 100 Hz. FFT was obtained from a 100 ms transient (10 ms to 110 ms) using a rectangular window (10-500Hz).

Table 5.1: FFT analysis results.

Parameter	Value
SNDR [dB]	43.3
SFDR [dBc]	52.58
ENOB [Bits]	6.9
SNR [dB]	43.71
THD [dB]	-52.74

5.5. EDO compensation

Figure 5.5 demonstrates the effectiveness of the BLC mechanism in handling the EDO present at the input. When the output crosses the upper threshold (TH+), D_{BLC} increases by one code, bringing the ADC output back to the linear region.

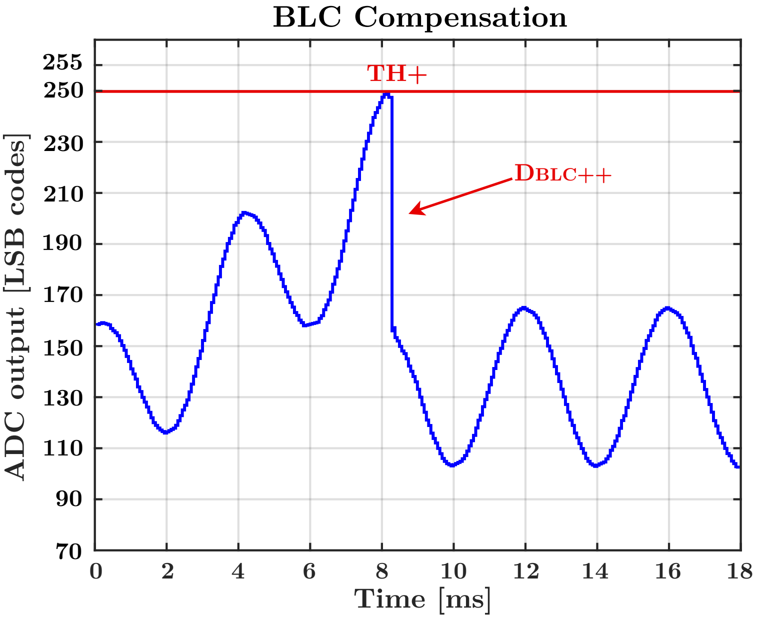
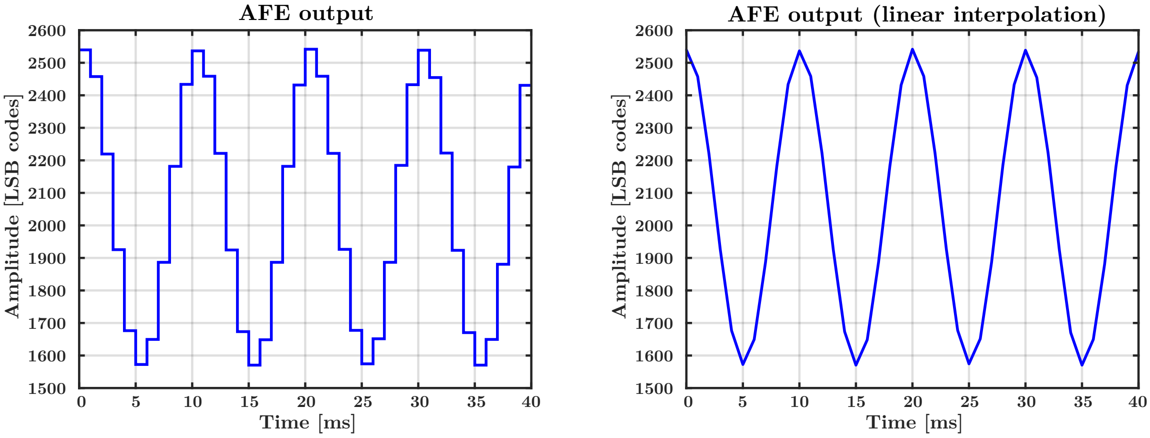


Figure 5.5: BLC compensation with an input signal of 1 mVpp at 250 Hz and an EDO increasing from 0 mV at 0 ms to 1.5 mV at 8 ms. The BLC upper threshold (TH+) is set to 11111001, equivalent to 249 LSB codes.

5.6. Signal reconstruction

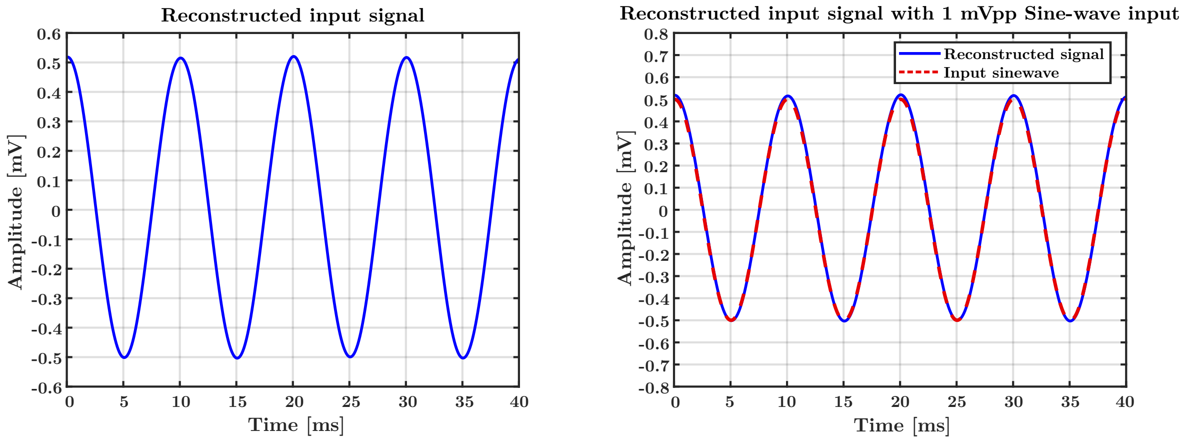
The output of the Analog Front End (AFE) when a 1 mV$_{pp}$ sine wave with a frequency of 100 Hz is applied to the input is shown in Figure 5.6. To reconstruct the original input signal, a spline interpolation is applied to the AFE output, and the result is scaled by the inverse of the measured gain (43.4 dB). Figure 5.7 illustrates the reconstructed signal and compares it with the original sine wave applied to the input.



(a) Original AFE output signal.

(b) AFE output signal with linear interpolation.

Figure 5.6: AFE output for a 1 mV$_{pp}$, 100 Hz sinusoidal input signal.



(a) Reconstructed input signal with a spline interpolation.

(b) Comparison of reconstructed and original input signals.

Figure 5.7: Reconstructed AFE input signal for a 1 mV$_{pp}$, 100 Hz sinusoidal input.

5.7. CMRR

The circuit maintains a constant Common-Mode Rejection Ratio (CMRR) of 80.94 dB in the signal bandwidth (Fig. 5.8).

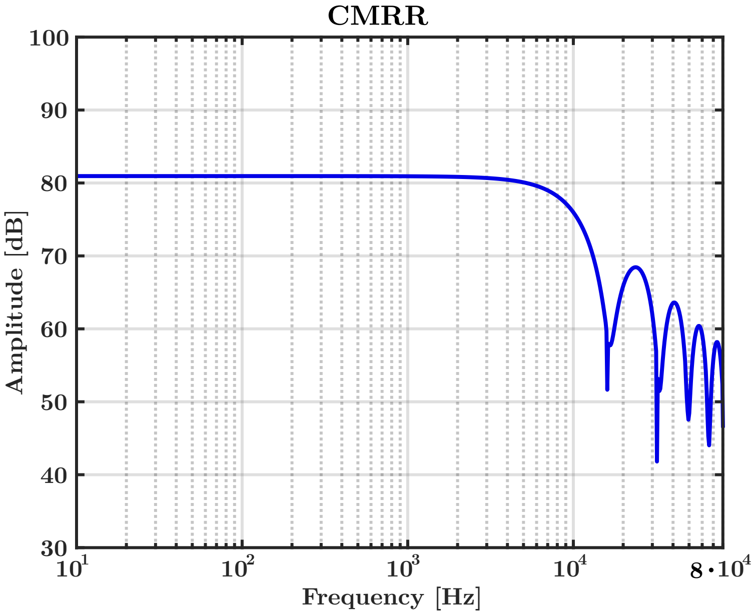


Figure 5.8: CMRR of the AFE.

5.8. Power consumption

The power breakdown for each of the main blocks of the AFE is exposed in this section.

5.8.1. OTA and CTC

The OTA (Operational Transconductance Amplifier) consumes 1.06 μW, while the CTC (Continuous-Time Comparator) uses 1.32 μW.

5.8.2. Bulk modulation chopping

The average power consumption related to the chopping of the BLC loop (for charging and discharging the body capacitances) can be derived as follows, considering the worst-case scenario where D_{BLC} is at its maximum, and C_{body} is set to the largest body capacitance for the MSB input device (12.23 fF).

$$\mathrm{P_{BLC,chopp}} = \mathrm{V_{DD,1.1\ V}} \cdot \frac{2 \cdot \Delta_{V_b} \cdot 12 \cdot C_{body}}{T_{chop}} = 3.71nW \tag{5.1}$$

5.8.3. Digital logic

The average power consumption of the digital blocks is summarized in Table 5.2.

Table 5.2: Average power consumption of different digital components.

Component	Average Power [nW]
Up-Dw (BLC logic)	12.03
Comparator (BLC logic)	72.83
Accumulator (OS)	8.58
Counter (ADC)	6.31
Total	99.76

5.8.4. Final power breakdown

Figure 5.9 presents the total power breakdown of the designed AFE. Components shared among all AFEs, such as the global ramp generator and the biasing circuitry, have their power contribution divided by the number of distributed AFEs in the array. Therefore, their impact is not dominant and is not considered in this analysis.

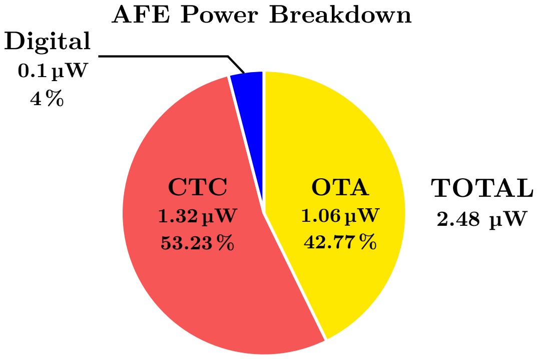


Figure 5.9: AFE power breakdown.

5.9. Comparison with Requirements

The obtained simulation results are exposed and compared with the initially set requirements in Table 5.3.

Table 5.3: Obtained performance parameters of the implemented AFE compared with the set requirements.

Parameter	Requirement	Obtained
Channels/AFE	1	1
Bandwidth	0.5 Hz - 500 Hz	0.5 Hz - 500 Hz
Noise	< 2 μV$_{rms}$	1.69 μV$_{rms}$
EDO Rejection	>100 mV_{pp}	110 mV_{pp}
Input impedance	>40 MΩ	71.4 MΩ
Power density	$< 1mW/mm^2$	0.88 mW/mm^2
CMRR	>75 dB	80.94 dB
Area/channel	$< 100\mu m \times 100\mu m$	$62\mu m \times 45\mu m$ [a]

[a] The area/channel provided is a first-order approximation based on all components of the AFE placed adjacently in a preliminary layout (Appendix D). This estimate does not account for routing, spacing requirements, or other layout considerations that may increase the final value. Moreover, shared components like the global ramp generator and biasing circuitry, distributed across multiple AFEs, are not considered due to their non-dominant contribution to individual AFE areas.

5.10. Comparison with the prior art

Table 5.4 presents the performance parameters of the designed AFE compared to the performance parameters of a selection of state-of-the-art μECoG readout systems.

The performance of the designed AFE is comparable to state-of-the-art systems, even those that leverage multiplexing techniques. It achieves low input-referred noise, competitive power consumption (only falling short compared to designs utilizing multiplexing techniques), and a high common-mode rejection ratio. Additionally, it provides a high input impedance while using CHS. It also demonstrates strong tolerance to EDO (which can be further improved as explained in section 3.1 and subsection 4.1.5), outperforming other systems in this regard.

The area efficiency of the proposed design is also noteworthy. With an estimated area of 0.0028 mm^2 per channel, it achieves a compact footprint that is smaller than most of the compared designs. This is evident when compared to other single-channel designs that don't employ multiplexing, such as SSCL'18 [84] (0.01 mm^2) and TBCaS'24 [52] (0.03 mm^2), resulting in improvements of 3.5x and 10.7x respectively. The area efficiency can be attributed to the elimination of additional input capacitors used in AC-coupled configurations and low-pass filters, and DACs typically required for the DSL implemented in other designs. It's important to note that this area estimate is preliminary and may increase slightly when developing the full layout. Nevertheless, the compact nature of the design positions it favorably for high-density neural recording applications.

Table 5.4: Comparison with prior art of μECoG readout circuits.

Reference	**This work**	**JSSC'15 [34]**	**SSCL'18 [84]**	**JSSC'22 [135]**	**JSSC'22 [29]**	**TBCaS'24 [52]**
Topology	DC Boxcar-SS ADC-BLC	AC AMP VCO-ADC	DC VCO-ADC-DSL	AC Boxcar-SAR ADC-DSL	I-$\Delta\Sigma$	DC I-ADC-DSL
TDM	NO	NO	NO	YES	YES	NO
Channels	1	64	1	256:4	256:16	1
Tech/V_{DD}	40nm/1.1V	65nm/0.5V	65nm/0.6V	65nm/1.2V	22nm/0.8V	180nm/1.8V
BW [Hz]	500	500	500	500	500	7500
IRN [µVrms]	1.69 (0.5-500 Hz)	1.2	2.2	3.2	1.55	1.46 (1-200 Hz)
Area [mm^2]	0.0028[a]	0.025	0.01	0.014	0.001	0.03
Power [µW/ch]	2.48	2.3	3.2	1.51	1.61	24.7
Input range [mVpp]	4	6	-	2.5	5	20
THD [dB]	-52.74 (1 mVpp)	-48 (1 mVpp)	-	-	-53 (2 mVpp)	-70.3 (20 mVpp)
CMRR [dB]	80.94	88	77	70	98	-
Zin [MΩ]	71.4 (100 Hz)	28 (100 Hz)	500	24.5 (100 Hz)	43	216
EDO Tol. [mVpp]	110	100	100	100	156.6	200

[a] The area/channel provided is a first-order approximation based on all components of the AFE placed adjacently in a preliminary layout. In addition, shared components like the global ramp generator and biasing circuitry, distributed across multiple AFEs, are not considered.

6

Future work and Conclusions

6.1. Future work

The AFE architecture has various possibilities for further research and development, which are discussed below.

6.1.1. Digital LPF

In this design, the D_{ADC} is accumulated 16 times, resulting in an equivalent rectangular window FIR digital filter. While this resulted in the simplest implementation, the effects of using other types of digital filters must also be studied and compared to choose the best possible solution.

6.1.2. BLC

The implementation of the BLC proposed in this thesis verified the feasibility of this concept. Additionally, it shows that the number of bits (resolution) or the compensation range could be further increased if needed. Therefore, it would be beneficial to verify this possibility for designs with different characteristics, to ensure the robustness of the presented compensation loop.

6.1.3. Power Consumption of the CTC

The CTC is responsible for over 50% of the total power consumption of this AFE. While the CTC was designed to meet the required bandwidth, the biasing current could be optimized to decrease the overall power consumption. Additionally, the implemented comparator draws current during the entire ADC conversion time, even after the ramp voltage and input voltage are equal, resulting in unnecessary power consumption. While a dynamic comparator may also not be a better choice, other architectures have been presented that aim to reduce this unnecessary power consumption in SS-ADCs [133]. However, no existing architecture is compatible with differential ramp signals, indicating the need for further development in this area.

6.1.4. Digital Logic

The digital logic in this thesis has been implemented with standard cells from the TSMC PDK and placed manually. For optimization in terms of power and area, Electronic Design Automation (EDA) tools could be utilized to refine the design. Additionally, different voltage supplies could be employed, with the digital logic being scaled down to further enhance efficiency.

6.1.5. Remaining Components and Layout

While the main components of the AFE have been implemented and their behavior verified, the low impedance bias generator used for setting the VB_{BLC} of the BLC, the global ramp generator circuit, and the on-chip clock generator based on delay lines also need to be studied and implemented. Additionally, the layout of the entire system should be completed to demonstrate the actual area benefits of the proposed architecture, particularly the proposed BLC.

6.1.6. Massively Parallel Array Compatibility

Although the proposed AFE is a compact solution for recording neural signals, it still requires multiple biasing signals. To reduce the routing constraints posed by a high channel count array, minimizing these biasing signals would be highly beneficial. The implementation of self-biased OTA and CTC architectures should be investigated to achieve this. Additionally, the proposed AFEs output a 12-bit digital word per sample, corresponding to 12 kbps per channel. Therefore, a feasible method for communicating all this data from each recording AFE to the central HUB must be developed. Furthermore, it is crucial to analyze which communication protocol would be most effective for sending control signals to each AFE.

6.2. Conclusions

This work aimed to propose a novel system architecture for recording neural signals, compatible with a massively parallel array. The previously implemented systems in the literature were carefully reviewed and served as a pre-study upon which this work is based. This study highlighted the best implementation for each part of the design, favoring an orthogonal and systematic approach.

The proposed AFE incorporates a novel bi-level bulk compensation (BLC) loop that attenuates the EDO present at the inputs without requiring input capacitors, thereby avoiding associated problems. It also eliminates the need for a DSL, which typically requires a DAC and a digital or analog low-pass filter. This novel compensation loop is directly embedded in the OTA, does not require additional hardware, and is compatible with standard CMOS technology. The architecture includes a DC-coupled chopper-stabilized low-noise boxcar sampler, a passive switched-capacitor low-pass filter, and a single-slope ADC, which together minimize noise folding and enhance anti-aliasing. Simulations validated the design's performance, demonstrating an IRN of 1.69 μV_{rms} over 0.5-500 Hz and an EDO compensation range above 100 mV_{pp}). Additionally, the design achieves an input impedance of 71.4 MΩ and a CMRR of 80.94 dB, meeting all the previously set requirements.

Notably, the proposed design exhibits exceptional area efficiency, with an estimated area of 0.0028 mm^2 per channel. This compact footprint represents a significant improvement over comparable single-channel designs. These results indicate that the presented AFE architecture, and specifically the novel compensation scheme, are a promising approach for neural recording applications, particularly in scenarios requiring high-density, massively parallel arrays.

Bibliography

[1] M. F. Mridha, S. C. Das, M. M. Kabir, A. A. Lima, M. R. Islam, and Y. Watanobe, "Brain-computer interface: Advancement and challenges," *Sensors*, vol. 21, no. 17, p. 5746, 2021.

[2] A. Mora-Cortes, N. V. Manyakov, N. Chumerin, and M. M. Van Hulle, "Language model applications to spelling with brain-computer interfaces," *Sensors*, vol. 14, no. 4, pp. 5967–5993, 2014.

[3] K. Belwafi, S. Gannouni, and H. Aboalsamh, "Embedded brain computer interface: State-of-the-art in research," *Sensors*, vol. 21, no. 13, p. 4293, 2021.

[4] K. Värbu, N. Muhammad, and Y. Muhammad, "Past, present, and future of eeg-based bci applications," *Sensors*, vol. 22, no. 9, p. 3331, 2022.

[5] J. Peksa and D. Mamchur, "State-of-the-art on brain-computer interface technology," *Sensors*, vol. 23, no. 13, p. 6001, 2023.

[6] S. Zanos, "Closed-loop neuromodulation in physiological and translational research," *Cold Spring Harbor perspectives in medicine*, vol. 9, no. 11, a034314, 2019.

[7] V. Asanza, E. Peláez, F. Loayza, L. L. Lorente-Leyva, and D. H. Peluffo-Ordóñez, "Identification of lower-limb motor tasks via brain–computer interfaces: A topical overview," *Sensors*, vol. 22, no. 5, p. 2028, 2022.

[8] N. Siribunyaphat and Y. Punsawad, "Brain–computer interface based on steady-state visual evoked potential using quick-response code pattern for wheelchair control," *Sensors*, vol. 23, no. 4, p. 2069, 2023.

[9] Y. Xie and S. Oniga, "Classification of motor imagery eeg signals based on data augmentation and convolutional neural networks," *Sensors*, vol. 23, no. 4, p. 1932, 2023.

[10] S. Niketeghad and N. Pouratian, "Brain machine interfaces for vision restoration: The current state of cortical visual prosthetics," *Neurotherapeutics*, vol. 16, no. 1, pp. 134–143, 2019.

[11] D. G. Muratore and E. Chichilnisky, "Artificial retina: A future cellular-resolution brain-machine interface," *NANO-CHIPS 2030: On-Chip AI for an Efficient Data-Driven World*, pp. 443–465, 2020.

[12] T. Milekovic, A. A. Sarma, D. Bacher, *et al.*, "Stable long-term bci-enabled communication in als and locked-in syndrome using lfp signals," *Journal of neurophysiology*, vol. 120, no. 7, pp. 343–360, 2018.

[13] S. Little, A. Pogosyan, S. Neal, *et al.*, "Adaptive deep brain stimulation in advanced parkinson disease," *Annals of neurology*, vol. 74, no. 3, pp. 449–457, 2013.

[14] G. Giuliana, M. Mario, and J. Yassin, "A quality parameter for the detection of the intentionality of movement in patients with neurological tremor performing a finger-to-nose test," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, 2011, pp. 7707–7710.

[15] H. Tayebi, S. Azadnajafabad, S. F. Maroufi, *et al.*, “Applications of brain-computer interfaces in neurodegenerative diseases,” *Neurosurgical Review*, vol. 46, no. 1, p. 131, 2023.

[16] R. Janapati, V. Dalal, N. Govardhan, and R. Sengupta, “Signal processing algorithms based on evolutionary optimization techniques in the bci: A review,” *Computational Vision and Bio-Inspired Computing: ICCVBIC 2020*, pp. 165–174, 2021.

[17] K. S. Park, “Electrical signal from the brain,” in *Humans and Electricity: Understanding Body Electricity and Applications*, Springer, 2023, pp. 173–197.

[18] R. Yuvaraj, P. Thagavel, J. Thomas, J. Fogarty, and F. Ali, “Comprehensive analysis of feature extraction methods for emotion recognition from multichannel eeg recordings,” *Sensors*, vol. 23, no. 2, p. 915, 2023.

[19] R. F. Yazicioglu, P. Merken, R. Puers, and C. Van Hoof, “A 200μ w eight-channel eeg acquisition asic for ambulatory eeg systems,” *IEEE Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 3025–3038, 2008.

[20] N. P. Prieto, “Low-power artifact-aware implantableneural recordingmicrosystem for brain-machine interfaces,” Ph.D. dissertation, Universidad de Sevilla, 2021.

[21] J. J. Jun, N. A. Steinmetz, J. H. Siegle, *et al.*, “Fully integrated silicon probes for high-density recording of neural activity,” *Nature*, vol. 551, no. 7679, pp. 232–236, 2017.

[22] N. A. Steinmetz, C. Aydin, A. Lebedeva, *et al.*, “Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings,” *Science*, vol. 372, no. 6539, eabf4588, 2021.

[23] J. C. Barrese, N. Rao, K. Paroo, *et al.*, “Failure mode analysis of silicon-based intracortical microelectrode arrays in non-human primates,” *Journal of neural engineering*, vol. 10, no. 6, p. 066 014, 2013.

[24] A. Prasad, Q.-S. Xue, V. Sankar, *et al.*, “Comprehensive characterization and failure modes of tungsten microwire arrays in chronic neural implants,” *Journal of neural engineering*, vol. 9, no. 5, p. 056 015, 2012.

[25] V. S. Polikov, P. A. Tresco, and W. M. Reichert, “Response of brain tissue to chronically implanted neural electrodes,” *Journal of neuroscience methods*, vol. 148, no. 1, pp. 1–18, 2005.

[26] R. W. Griffith and D. R. Humphrey, “Long-term gliosis around chronically implanted platinum electrodes in the rhesus macaque motor cortex,” *Neuroscience letters*, vol. 406, no. 1-2, pp. 81–86, 2006.

[27] M. Shokoueinejad, D.-W. Park, Y. H. Jung, *et al.*, “Progress in the field of micro-electrocorticography,” *Micromachines*, vol. 10, no. 1, p. 62, 2019.

[28] S. Kellis, L. Sorensen, F. Darvas, *et al.*, “Multi-scale analysis of neural activity in humans: Implications for micro-scale electrocorticography,” *Clinical Neurophysiology*, vol. 127, no. 1, pp. 591–601, 2016.

[29] X. Huang, H. Londoño-Ramírez, M. Ballini, *et al.*, "Actively multiplexed μecog brain implant system with incremental-$\Delta\Sigma$ adcs employing bulk-dacs," *IEEE Journal of Solid-State Circuits*, vol. 57, no. 11, pp. 3312–3323, 2022.

[30] D. Khodagholy, J. N. Gelinas, T. Thesen, *et al.*, "Neurogrid: Recording action potentials from the surface of the brain," *Nature neuroscience*, vol. 18, no. 2, pp. 310–315, 2015.

[31] S. Brincat, N. Jia, A. Salazar-Gómez, M. Panko, E. Miller, and F. Guenther, "Which neural signals are optimal for brain-computer interface control," in *Proceedings of the Fifth International Brain-Computer Interface Meeting*, 2013, p. 104.

[32] A. Jackson and T. M. Hall, "Decoding local field potentials for neural interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1705–1714, 2016.

[33] D. A. Heldman and D. W. Moran, "Local field potentials for bci control," *Handbook of clinical neurology*, vol. 168, pp. 279–288, 2020.

[34] R. Muller, H.-P. Le, W. Li, *et al.*, "A minimally invasive 64-channel wireless μecog implant," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 344–359, 2014.

[35] A. Akinin, A. Paul, J. Wang, A. Buccino, and G. Cauwenberghs, "Biopotential measurements and electrodes," *Neural engineering*, pp. 65–96, 2020.

[36] J. Guo, J. Yuan, and M. Chan, "Modeling of the cell-electrode interface noise for microelectrode arrays," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 6, no. 6, pp. 605–613, 2012.

[37] M. Maslik, L. B. Leene, and T. G. Constandinou, "Analogue front-end design for neural recording," *Handbook of Neuroengineering*, pp. 1–26, 2020.

[38] F. Hashemi Noshahr, M. Nabavi, and M. Sawan, "Multi-channel neural recording implants: A review," *sensors*, vol. 20, no. 3, p. 904, 2020.

[39] A. Bagheri, M. T. Salam, J. L. P. Velazquez, and R. Genov, "Low-frequency noise and offset rejection in dc-coupled neural amplifiers: A review and digitally-assisted design tutorial," *IEEE transactions on biomedical circuits and systems*, vol. 11, no. 1, pp. 161–176, 2016.

[40] Y. Xie, Y. Peng, J. Guo, *et al.*, "Materials and devices for high-density, high-throughput micro-electrocorticography arrays," *Fundamental Research*, 2024.

[41] W. M. Reichert, *Indwelling neural implants: strategies for contending with the in vivo environment.* CRC Press, 2007.

[42] K. Sharma and R. Sharma, "Design considerations for effective neural signal sensing and amplification: A review," *Biomedical Physics & Engineering Express*, vol. 5, no. 4, p. 042 001, 2019.

[43] M. K. Kim, H. Jeon, H. J. Lee, and M. Je, "Plugging electronics into minds: Recent trends and advances in neural interface microsystems," *IEEE Solid-State Circuits Magazine*, vol. 11, no. 4, pp. 29–42, 2019.

[44] I. H. Stevenson and K. P. Kording, "How advances in neural recording affect data analysis," *Nature neuroscience*, vol. 14, no. 2, pp. 139–142, 2011.

[45] W. Lee, D. Kim, N. Matsuhisa, *et al.*, “Transparent, conformable, active multielectrode array using organic electrochemical transistors,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 40, pp. 10 554–10 559, 2017.

[46] J. Viventi, D.-H. Kim, L. Vigeland, *et al.*, “Flexible, foldable, actively multiplexed, high-density electrode array for mapping brain activity in vivo,” *Nature neuroscience*, vol. 14, no. 12, pp. 1599–1605, 2011.

[47] C.-H. Chiang, S. M. Won, A. L. Orsborn, *et al.*, “Development of a neural interface for high-definition, long-term recording in rodents and nonhuman primates,” *Science translational medicine*, vol. 12, no. 538, eaay4682, 2020.

[48] N. Pérez-Prieto and M. Delgado-Restituto, “Recording strategies for high channel count, densely spaced microelectrode arrays,” *Frontiers in Neuroscience*, vol. 15, p. 681 085, 2021.

[49] N. S. K. Fathy, J. Huang, and P. P. Mercier, “A digitally assisted multiplexed neural recording system with dynamic electrode offset cancellation via an lms interference-canceling filter,” *IEEE Journal of Solid-State Circuits*, vol. 57, no. 3, pp. 953–964, 2021.

[50] R. M. Walker, L. Rieth, S. S. Iyer, *et al.*, “Integrated neural interfaces,” in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, IEEE, 2017, pp. 1045–1048.

[51] M. Jang *et al.*, “A 1024-channel 268-nw/pixel 36×36 μm^2/channel data-compressive neural recording ic for high-bandwidth brain–computer interfaces,” *IEEE Journal of Solid-State Circuits*, vol. 59, no. 4, pp. 1123–1136, Apr. 2024.

[52] M. Sporer *et al.*, “Neurobus - architecture for an ultra-flexible neural interface,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 18, no. 2, pp. 247–262, Apr. 2024.

[53] R. R. FULLY INTEGRA and C. Charles, “A low-power low-noise cmos amplifier for neural recording applications,” *IEEE Journal of solid-state circuits*, vol. 38, no. 6, pp. 958–965, 2003.

[54] W. Wattanapanitch, M. Fee, and R. Sarpeshkar, “An energy-efficient micropower neural recording amplifier,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 1, no. 2, pp. 136–147, 2007.

[55] X. Zou, X. Xu, L. Yao, and Y. Lian, “A 1-v 450-nw fully integrated programmable biomedical sensor interface chip,” *IEEE journal of solid-state circuits*, vol. 44, no. 4, pp. 1067–1077, 2009.

[56] R. R. FULLY INTEGRA, P. T. Watkins, R. J. Kier, *et al.*, “A low-power integrated circuit for a wireless 100-electrode neural recording system,” *IEEE Journal of Solid-State Circuits*, vol. 42, no. 1, pp. 123–133, 2006.

[57] J. Guo, W. Ng, J. Yuan, S. Li, and M. Chan, “A 200-channel area-power-efficient chemical and electrical dual-mode acquisition ic for the study of neurodegenerative diseases,” *IEEE transactions on biomedical circuits and systems*, vol. 10, no. 3, pp. 567–578, 2015.

[58] T. Denison, K. Consoer, W. Santa, A.-T. Avestruz, J. Cooley, and A. Kelly, "A 2μW 100 nv/$\sqrt{\text{Hz}}$ chopper-stabilized instrumentation amplifier for chronic measurement of neural field potentials," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 12, pp. 2934–2945, 2007.

[59] H. Song, Y. Park, H. Kim, and H. Ko, "Fully integrated biopotential acquisition analog front-end ic," *Sensors*, vol. 15, no. 10, pp. 25 139–25 156, 2015.

[60] L. Liu, L. Yao, X. Zou, W. L. Goh, and M. Je, "Neural recording front-end ic using action potential detection and analog buffer with digital delay for data compression," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2013, pp. 747–750.

[61] S. Schmickl, T. Schumacher, P. Fath, T. Faseth, and H. Pretl, "A 350-nw low-noise amplifier with reduced flicker-noise for bio-signal acquisition," in *2020 austrochip workshop on microelectronics (austrochip)*, IEEE, 2020, pp. 9–12.

[62] G. Gagnon-Turcotte, M. N. Khiarak, C. Ethier, Y. De Koninck, and B. Gosselin, "A 0.13-μm cmos soc for simultaneous multichannel optogenetics and neural recording," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 11, pp. 3087–3100, 2018.

[63] K. A. Ng and Y. P. Xu, "A compact, low input capacitance neural recording amplifier," *IEEE Transactions on biomedical circuits and systems*, vol. 7, no. 5, pp. 610–620, 2013.

[64] M. M. Ghanbari, D. K. Piech, K. Shen, *et al.*, "17.5 a 0.8 mm 3 ultrasonic implantable wireless neural recording system with linear am backscattering," in *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, IEEE, 2019, pp. 284–286.

[65] H. Gao, R. M. Walker, P. Nuyujukian, *et al.*, "Hermese: A 96-channel full data rate direct neural interface in 0.13μm cmos," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 4, pp. 1043–1055, 2012.

[66] J. Dragas, V. Viswam, A. Shadmani, *et al.*, "In vitro multi-functional microelectrode array featuring 59 760 electrodes, 2048 electrophysiology channels, stimulation, impedance measurement, and neurotransmitter detection channels," *IEEE journal of solid-state circuits*, vol. 52, no. 6, pp. 1576–1590, 2017.

[67] N. Laskar, K. Guha, S. Nath, *et al.*, "Design of high gain, high bandwidth neural amplifier ic considering noise-power trade-off," *Microsystem technologies*, vol. 27, pp. 585–599, 2021.

[68] E. Guglielmi, F. Toso, F. Zanetto, *et al.*, "High-value tunable pseudo-resistors design," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 8, pp. 2094–2105, 2020.

[69] M.-T. Shiue, K.-W. Yao, and C.-S. A. Gong, "A bandwidth-tunable bioamplifier with voltage-controlled symmetric pseudo-resistors," *Microelectronics Journal*, vol. 46, no. 6, pp. 472–481, 2015.

[70] P. Mohseni and K. Najafi, "A fully integrated neural recording amplifier with dc input stabilization," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 5, pp. 832–837, 2004.

[71] H. Rezaee-Dehsorkh, N. Ravanshad, R. Lotfi, K. Mafinezhad, and A. M. Sodagar, "Analysis and design of tunable amplifiers for implantable neural recording applications," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 4, pp. 546–556, 2011.

[72] F. Zhang, A. Mishra, A. G. Richardson, and B. Otis, "A low-power ecog/eeg processing ic with integrated multiband energy extractor," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 9, pp. 2069–2082, 2011.

[73] E. C. M. Lim, X. Zou, Y. Zheng, and J. Tan, "Design of low-power low-voltage biomedical amplifier for electrocardiogram signal recording," in *2007 IEEE Biomedical Circuits and Systems Conference*, IEEE, 2007, pp. 191–194.

[74] R. H. Olsson, D. L. Buhl, A. M. Sirota, G. Buzsáki, and K. D. Wise, "Band-tunable and multiplexed integrated circuits for simultaneous recording and stimulation with microelectrode arrays," *IEEE transactions on biomedical engineering*, vol. 52, no. 7, pp. 1303–1311, 2005.

[75] H. Chandrakumar and D. Marković, "A high dynamic-range neural recording chopper amplifier for simultaneous neural recording and stimulation," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 3, pp. 645–656, 2017.

[76] W. Jiang, V. Hokhikyan, H. Chandrakumar, V. Karkare, and D. Marković, "A±50-mv linear-input-range vco-based neural-recording front-end with digital nonlinearity correction," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, pp. 173–184, 2016.

[77] Y.-K. Huang and S. Rodriguez, "Noise analysis and design methodology of chopper amplifiers with analog dc-servo loop for biopotential acquisition applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2023.

[78] R. F. Yazicioglu, "A 60μw 60 nv/$\surd$< hz> readout front-end for portablebiopotential acquisition systems," in *IEEE International Solid-State Circuits Conference Digest of Technical Papers, Feb. 2006*, 2006.

[79] B. Gosselin, M. Sawan, and C. A. Chapman, "A low-power integrated bioamplifier with active low-frequency suppression," *IEEE Transactions on biomedical circuits and systems*, vol. 1, no. 3, pp. 184–192, 2007.

[80] M. Sporer, S. Reich, J. G. Kauffman, and M. Ortmanns, "A direct digitizing chopped neural recorder using a body-induced offset based dc servo loop," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 16, no. 3, pp. 409–418, 2022.

[81] A. Bagheri, M. T. Salam, J. L. P. Velazquez, and R. Genov, "56-channel direct-coupled chopper-stabilized eeg monitoring asic with digitally-assisted offset correction at the folding nodes," in *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*, IEEE, 2014, pp. 659–662.

[82] R. Muller, S. Gambini, and J. M. Rabaey, "A 0.013 mm^2, 5 μW, dc-coupled neural signal acquisition ic with 0.5 V supply," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 1, pp. 232–243, 2011.

[83] T. Moeinfard, G. Zoidl, and H. Kassiri, “A sar-assisted dc-coupled chopper-stabilized 20μs-artifact-recovery $\Delta\Sigma$ adc for simultaneous neural recording and stimulation,” in *2022 IEEE Custom Integrated Circuits Conference (CICC)*, IEEE, 2022, pp. 1–2.

[84] J. Huang, F. Laiwalla, J. Lee, *et al.*, “A 0.01-mm 2 mostly digital capacitor-less afe for distributed autonomous neural sensor nodes,” *IEEE Solid-State Circuits Letters*, vol. 1, no. 7, pp. 162–165, 2018.

[85] H. Jeon, J.-S. Bang, Y. Jung, I. Choi, and M. Je, “A high dr, dc-coupled, time-based neural-recording ic with degeneration r-dac for bidirectional neural interface,” *IEEE Journal of Solid-State Circuits*, vol. 54, no. 10, pp. 2658–2670, 2019.

[86] C. Lee, T. Jeon, M. Jang, *et al.*, “A 6.5-μw 10-khz bw 80.4-db sndr g m-c-based ct modulator with a feedback-assisted g m linearization for artifact-tolerant neural recording,” *IEEE Journal of Solid-State Circuits*, vol. 55, no. 11, pp. 2889–2901, 2020.

[87] X. T. Pham, N. T. Nguyen, V. T. Nguyen, and J.-W. Lee, “A 0.6-μw chopper amplifier using a noise-efficient dc servo loop and squeezed-inverter stage for power-efficient biopotential sensing,” *Sensors*, vol. 20, no. 7, p. 2059, 2020.

[88] B. Razavi, “Design of analog cmos integrated circuits,” *Design of Analog CMOS Integrated Circuits*, 2003.

[89] N. Van Helleputte, C. Mora-Lopez, and C. Van Hoof, “Design of cmos circuits for electrophysiology,” *IEICE Transactions on Electronics*, vol. 106, no. 10, pp. 506–515, 2023.

[90] J. L. Valtierra, R. Fiorelli, M. Delgado-Restituto, and A. Rodriguez-Vazquez, “A sub-μw reconfigurable front-end for invasive neural recording,” in *2019 IEEE 10th Latin American Symposium on Circuits & Systems (LASCAS)*, IEEE, 2019, pp. 85–88.

[91] T. Denison, K. Consoer, W. Santa, A.-T. Avestruz, J. Cooley, and A. Kelly, “A 2 μW 100 nv/$\sqrt{\text{Hz}}$ chopper-stabilized instrumentation amplifier for chronic measurement of neural field potentials,” *IEEE journal of solid-state circuits*, vol. 42, no. 12, pp. 2934–2945, 2007.

[92] J. Xu, R. F. Yazicioglu, B. Grundlehner, P. Harpe, K. A. Makinwa, and C. Van Hoof, “A 160 μ\W 8-channel active electrode system for eeg monitoring,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 5, no. 6, pp. 555–567, 2011.

[93] N. Verma, A. Shoeb, J. Bohorquez, J. Dawson, J. Guttag, and A. P. Chandrakasan, “A micro-power eeg acquisition soc with integrated feature extraction processor for a chronic seizure detection system,” *IEEE journal of solid-state circuits*, vol. 45, no. 4, pp. 804–816, 2010.

[94] Q. Fan, F. Sebastiano, J. H. Huijsing, and K. A. Makinwa, “A 1.8 μw 60 nv/$\sqrt{\text{Hz}}$ capacitively-coupled chopper instrumentation amplifier in 65 nm cmos for wireless sensor nodes,” *IEEE Journal of Solid-State Circuits*, vol. 46, no. 7, pp. 1534–1543, 2011.

[95] J. Liu and D. J. Allstot, “Windowed integration sampling in bio-signal front-end design,” in *2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS)*, IEEE, 2020, pp. 529–532.

[96] A. Mirzaei, S. Chehrazi, R. Bagheri, and A. A. Abidi, “Analysis of first-order anti-aliasing integration sampler,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 10, pp. 2994–3005, 2008.

[97] X. Ge and A. J. Theuwissen, “Temporal noise analysis of charge-domain sampling readout circuits for cmos image sensors,” *Sensors*, vol. 18, no. 3, p. 707, 2018.

[98] G. Xu and J. Yuan, “Comparison of charge sampling and voltage sampling,” in *Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems (Cat. No. CH37144)*, IEEE, vol. 1, 2000, pp. 440–443.

[99] J. L. Bohorquez, M. Yip, A. P. Chandrakasan, and J. L. Dawson, “A biomedical sensor interface with a sinc filter and interference cancellation,” *IEEE Journal of Solid-State Circuits*, vol. 46, no. 4, pp. 746–756, 2011.

[100] B. Razavi, “The delta-sigma modulator [a circuit for all seasons],” *IEEE Solid-State Circuits Magazine*, vol. 8, no. 2, pp. 10–15, 2016.

[101] J.-S. Bang, H. Jeon, M. Je, and G.-H. Cho, “6.5 μw 92.3 db-dr biopotential-recording front-end with 360mv pp linear input range,” in *2018 IEEE Symposium on VLSI Circuits*, IEEE, 2018, pp. 239–240.

[102] H. Chandrakumar and D. Marković, “A 15.2-enob 5-khz bw 4.5-μw chopped ct $\Delta\Sigma$-adc for artifact-tolerant neural recording front ends,” *IEEE Journal of Solid-State Circuits*, vol. 53, no. 12, pp. 3470–3483, 2018.

[103] H. Jeon, J.-S. Bang, Y. Jung, I. Choi, and M. Je, “A high dr, dc-coupled, time-based neural-recording ic with degeneration r-dac for bidirectional neural interface,” *IEEE Journal of Solid-State Circuits*, vol. 54, no. 10, pp. 2658–2670, 2019.

[104] E. Greenwald, E. So, Q. Wang, *et al.*, “A bidirectional neural interface ic with chopper stabilized bioadc array and charge balanced stimulator,” *IEEE transactions on biomedical circuits and systems*, vol. 10, no. 5, pp. 990–1002, 2016.

[105] M. R. Pazhouhandeh, M. Chang, T. A. Valiante, and R. Genov, “Track-and-zoom neural analog-to-digital converter with blind stimulation artifact rejection,” *IEEE Journal of Solid-State Circuits*, vol. 55, no. 7, pp. 1984–1997, 2020.

[106] M. R. Pazhouhandeh, H. Kassiri, A. Shoukry, I. Weisspapir, P. L. Carlen, and R. Genov, “Opamp-less sub-μw/channel Δ-modulated neural-adc with super-gΩ input impedance,” *IEEE Journal of Solid-State Circuits*, vol. 56, no. 5, pp. 1565–1575, 2020.

[107] S. Wang, M. Ballini, X. Yang, *et al.*, “A compact chopper stabilized Δ-$\Delta\Sigma$ neural readout ic with input impedance boosting,” *IEEE Open Journal of the Solid-State Circuits Society*, vol. 1, pp. 67–78, 2021.

[108] X. Huang, M. Ballini, S. Wang, *et al.*, “A compact, low-power analog front-end with event-driven input biasing for high-density neural recording in 22-nm fdsoi,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 3, pp. 804–808, 2021.

[109] D. Wendler, D. De Dorigo, M. Amayreh, *et al.*, "A 0.0046-mm 2 two-step incremental delta–sigma analog-to-digital converter neuronal recording front end with 120-mvpp offset compensation," *IEEE Journal of Solid-State Circuits*, vol. 58, no. 2, pp. 439–450, 2022.

[110] X. Yang, M. Ballini, C. Sawigun, *et al.*, "A 128-channel ac-coupled 1 st-order Δ-Δ ic for neural signal acquisition," in *2022 IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits)*, IEEE, 2022, pp. 60–61.

[111] W. Smith, J. Uehlin, S. Perlmutter, J. Rudell, and V. Sathe, "A scalable, highly-multiplexed delta-encoded digital feedback ecog recording amplifier with common and differential-mode artifact suppression," in *2017 Symposium on VLSI Circuits*, IEEE, 2017, pp. C172–C173.

[112] H. Kassiri, R. Pazhouhandeh, N. Soltani, *et al.*, "27.3 all-wireless 64-channel 0.013 mm 2/ch closed-loop neurostimulator with rail-to-rail dc offset removal," in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, IEEE, 2017, pp. 452–453.

[113] C.-H. Chan, Y. Zhu, S.-W. Sin, U. Seng-Pan, and R. P. Martins, "A 3.8 mw 8b 1gs/s 2b/cycle interleaving sar adc with compact dac structure," in *2012 Symposium on VLSI Circuits (VLSIC)*, IEEE, 2012, pp. 86–87.

[114] H. Li, Y. Shen, E. Cantatore, and P. Harpe, "Small-area sar adcs with a compact unit-length dac layout," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 10, pp. 4038–4042, 2022.

[115] B. P. Ginsburg and A. P. Chandrakasan, "An energy-efficient charge recycling approach for a sar converter with capacitive dac," in *2005 IEEE international symposium on circuits and systems*, IEEE, 2005, pp. 184–187.

[116] X. Tong and M. Ghovanloo, "Energy-efficient switching scheme in sar adc for biomedical electronics," *Electronics Letters*, vol. 51, no. 9, pp. 676–678, 2015.

[117] X. Tong, W. Zhang, and F. Li, "Low-energy and area-efficient switching scheme for sar a/d converter," *Analog Integrated Circuits and Signal Processing*, vol. 80, pp. 153–157, 2014.

[118] Y. Zhang, C. Yang, J. Sun, *et al.*, "A wireless headstage system based on neural-recording chip featuring 315 nw kickback-reduction sar adc," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 17, no. 1, pp. 105–115, 2022.

[119] Y. Khazaei and A. M. Sodagar, "Multi-channel adc with improved bit rate and power consumption for electrocorticography systems," in *2019 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, IEEE, 2019, pp. 1–4.

[120] W. Smith, J. Uehlin, S. Perlmutter, J. Rudell, and V. Sathe, "A scalable, highly-multiplexed delta-encoded digital feedback ecog recording amplifier with common and differential-mode artifact suppression," in *2017 Symposium on VLSI Circuits*, IEEE, 2017, pp. C172–C173.

[121] X. Huang, M. Ballini, S. Wang, *et al.*, "A compact, low-power analog front-end with event-driven input biasing for high-density neural recording in 22-nm fdsoi," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 3, pp. 804–808, 2021.

[122] R. Narwaria and M. Kumar, "Comparison of fir filter using different window functions," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 10, pp. 2395–0056, 2018.

[123] S. Zahoor and S. Naseem, "Design and implementation of an efficient fir digital filter," *Cogent Engineering*, vol. 4, no. 1, p. 1 323 373, 2017.

[124] L. Lyu, Y. Wang, C. Chen, and C.-J. R. Shi, "A 0.6 v 1.07 μw/channel neural interface ic using level-shifted feedback," *Integration*, vol. 70, pp. 51–59, 2020.

[125] P. G. Jespers and B. Murmann, *Systematic design of analog CMOS circuits*. Cambridge University Press, 2017.

[126] M. J. Pelgrom, A. C. Duinmaijer, and A. P. Welbers, "Matching properties of mos transistors," *IEEE Journal of solid-state circuits*, vol. 24, no. 5, pp. 1433–1439, 1989.

[127] D. Khodagholy, J. N. Gelinas, and G. Buzsáki, "Learning-enhanced coupling between ripple oscillations in association cortices and hippocampus," *Science*, vol. 358, no. 6361, pp. 369–372, 2017.

[128] H. Londoño-Ramirez, X. Huang, J. Cools, *et al.*, "Multiplexed surface electrode arrays based on metal oxide thin-film electronics for high-resolution cortical mapping," *Advanced Science*, vol. 11, no. 10, p. 2 308 507, 2024.

[129] P. E. Allen and D. R. Holberg, *CMOS analog circuit design*. Elsevier, 2011.

[130] Y. Kusuda, "Reducing switching artifacts in chopper amplifiers," 2018.

[131] O. Choksi and L. R. Carley, "Analysis of switched-capacitor common-mode feedback circuit," *IEEE Transactions on Circuits and Systems II: Analog and digital signal processing*, vol. 50, no. 12, pp. 906–917, 2003.

[132] B. Wang, S. Wang, and M.-K. Law, "On low-leakage cmos switches," in *2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, IEEE, 2021, pp. 1–5.

[133] W. Jendernalik, "An ultra-low-energy analog comparator for a/d converters in cmos image sensors," *Circuits, Systems, and Signal Processing*, vol. 36, no. 12, pp. 4829–4843, 2017.

[134] B. Razavi, "The design of a comparator [the analog mind]," *IEEE Solid-State Circuits Magazine*, vol. 12, no. 4, pp. 8–14, 2020.

[135] U. Shin, L. Somappa, C. Ding, *et al.*, "A 256-channel 0.227 μj/class versatile brain activity classification and closed-loop neuromodulation soc with 0.004 mm 2-1.51 μw/channel fast-settling highly multiplexed mixed-signal front-end," in *2022 IEEE International Solid-State Circuits Conference (ISSCC)*, IEEE, vol. 65, 2022, pp. 338–340.

A

Matlab codes

This Appendix provides the Matlab codes used to design or verify the proposed AFE.

A.1. Folded Cascode

Code used to bias and size the folded cascode OTA by the GM/ID methodology,

```
FOLDED CASCODE IMPLEMENTATION
%In this stage, the GM/ID methodology will be implemented to ensure a systematic design approach. The primary objective of this design is to minimize the Integrated Referred Noise (IRN) and power consumption. Therefore, the current distribution is as follows: the output branch current is set to only 1/10th of the input current.}

%To avoid flicker noise, chopping will be performed on the system, making thermal noise the main contributor to the circuit. The input transistors and the lower current sources are the primary sources of thermal noise, as the cascodes do not contribute to the noise, and the top current sources have much lower current, resulting in lower transconductance (gm).


%INITIAL CONDITIONS
Kb=1.38E-23;
T=300;
BW=500;

input_biasing_current=0.8e-6;
gamma=1; %To have some margin)
Current_input=input_biasing_current/2;
Current_output=0.1*input_biasing_current;

GM_ID_1=25; %Weak inversion
GM_ID_2=15; %Moderate inversion
gm_1=GM_ID_1*Current_input %This is the gm of the input transistors of the ota
gm_2=GM_ID_2*(Current_input+Current_output) %Input-referred noise

Vint=sqrt(8*Kb*T*((gamma/gm_1)+(gamma*(gm_2/gm_1^2)))*BW)

%INPUT BRANCH
%For the input transistor, it is needed to set the VDS.

load 40pch_25.mat %Thick oxide
L_INPUT=pch.L;
```

```matlab
VGS_1 = look_upVGS(pch,'GM_ID',GM_ID_1,'L',L_INPUT);
ro_in = 1./(gm_1./look_up(pch,  'GM_GDS', 'GM_ID', GM_ID_1, 'VDS', VGS_1-0.2, 'L', L_INPUT));
ID_W_1 = look_up(pch, 'ID_W', 'GM_ID', GM_ID_1, 'L', L_INPUT, 'VDS', VGS_1-0.2);
W_1= Current_input./ID_W_1;


% Plot VGS vs L
figure;
plot(L_INPUT, VGS_1);
xlabel('Input transistors length [um]');
ylabel('VGS of input transistor [um]');
title('Vgs vs length of input transistors');
grid on;

% Plot W_top vs L_INPUT
figure;
plot(L_INPUT, W_1);
xlabel('Input transistors length [um]');
ylabel('Input transistors width [um]');
title('Width vs length of input transistors');
grid on;

% Plot ro_in vs L_INPUT
figure;
plot(L_INPUT, ro_in*1e-6);
xlabel('Input transistors length [um]');
ylabel('Output resistance of input transistors [MOhm]');
title('Output impedance vs length of input transistor');
grid on;

%Now for the top transistor
GM_ID_TOP=10;

%Loop to verify which L to set both to the input transistor and the top transistor

load 40pch_svt.mat
Length_top=pch.L;
% Initialize matrices to store values for plotting
ro_top_all = zeros(length(Length_top), length(VGS_1));
W_top_all = zeros(length(Length_top), length(VGS_1));

% Loop over all VGS_1 values
for k = 1:length(VGS_1)
    VGS = VGS_1(k);
    for i = 1:length(Length_top)
        L = Length_top(i);
        ro_top = 1 ./ (gm_2 ./ look_up(pch, 'GM_GDS', 'GM_ID', GM_ID_TOP, 'VDS', 1.1 - VGS, 'L', L));
        ID_W_top = look_up(pch, 'ID_W', 'GM_ID', GM_ID_TOP, 'L', L, 'VDS', 1.1 - VGS);
        W_1 = Current_input * 2 ./ ID_W_top;

        ro_top_all(i, k) = ro_top;
        W_top_all(i, k) = W_1;
    end
end

% Plot ro_top vs Length_top for each VGS_1 value
figure;
hold on;
for k = 1:length(VGS_1)
    plot(Length_top, ro_top_all(:, k), 'DisplayName', ['VGS = ' num2str(VGS_1(k))]);
end
hold off;
xlabel('Length (L)');
ylabel('ro\_top');
title('ro\_top vs L for different VGS\_1 values');
legend('show');
grid on;

% Plot W_top vs Length_top for each VGS_1 value
figure;
hold on;
for k = 1:length(VGS_1)
    plot(Length_top, W_top_all(:, k), 'DisplayName', ['VGS = ' num2str(VGS_1(k))]);
end
hold off;
xlabel('Length (L)');
ylabel('W\_top');
title('W\_top vs L for different VGS\_1 values');
legend('show');
grid on;



%Considering the input transistors will be binary splitted a Length of 1 um is chosen to ensure mismatch resilience, The higher the resistance of the top current source Rtail, the lower the common mode gain and the better the common mode rejection ratio (CMRR).


load 40pch_25.mat
VGS_1 = look_upVGS(pch,'GM_ID',GM_ID_1,'L',1)
ID_W_1 = look_up(pch, 'ID_W', 'GM_ID', GM_ID_1, 'L', 1, 'VDS', VGS_1-0.2);
W_1= Current_input./ID_W_1
ro_in = 1./(gm_1./look_up(pch,  'GM_GDS', 'GM_ID', GM_ID_1, 'VDS', VGS_1-0.2, 'L', 1));

%Then, for the top transistor we can choose a length of 0.5 um.
load 40pch_svt.mat
L_TOP=1;
ID_W_top = look_up(pch, 'ID_W', 'GM_ID', GM_ID_TOP, 'L', 2, 'VDS', 1.1 - VGS_1);
W_TOP = Current_input * 2 ./ ID_W_top
```

```
 

Output branch

GM_ID_5= 15; %Current source top
GM_ID_4= 20; %Cascode low
GM_ID_3= 20; %Cascode top
GM_ID_2= 15; %Current source low


load 40nch_svt.mat
load 40pch_svt.mat
%To set the length of the ouput transistors, it is needed to check the output impedance of the branch as it will effectively set the gain of the boxcar.

L=pch.L;
%To set the gms of the transistors at the output branch:
gm_1=GM_ID_1*Current_input %This is the gm of the input transistors of the ota
gm_2=GM_ID_2*(Current_input+Current_output);
gm_3=GM_ID_3*Current_output;
gm_4=GM_ID_4*Current_output;
gm_5=GM_ID_5*Current_output

L = pch.L;

% Define Current_input, Current_output, GM_ID_1 to GM_ID_5, and ID1_opt if not already done.
% Assumed ro_in is already defined as a scalar.
% Pre-allocate arrays based on L's length
ro_2 = zeros(size(L));
ro_3 = zeros(size(L));
ro_4 = zeros(size(L));
ro_5 = zeros(size(L));

% Calculation of ro_X for each transistor
for i = 1:length(L)
    ro_2(i) = 1 / (gm_2 / look_up(nch, 'GM_GDS', 'GM_ID', 15, 'VDS', 0.2, 'L', L(i)));
    ro_3(i) = 1 / (gm_3 / look_up(nch, 'GM_GDS', 'GM_ID', 20, 'VDS', 0.35, 'L', L(i)));
    ro_4(i) = 1 / (gm_4 / look_up(pch, 'GM_GDS', 'GM_ID', 20, 'VDS', 0.35, 'L', L(i)));
    ro_5(i) = 1 / (gm_5 / look_up(pch, 'GM_GDS', 'GM_ID', 15, 'VDS', 0.2, 'L', L(i)));
end

% Calculate the OTA's output resistance without needing to loop, since ro_in is a scalar
% Calculate parallel_ro2_ro5 using a for loop
parallel_ro2_roin = zeros(size(L));
for i = 1:length(L)
    parallel_ro2_roin_1 = (ro_2(i) * ro_in);
    parallel_ro2_roin_2 = (ro_2(i) + ro_in);
    parallel_ro2_roin(i)=parallel_ro2_roin_1/parallel_ro2_roin_2;
end

lower_casc_ro = (parallel_ro2_roin .* ro_3) * gm_3;
upper_casc_ro = (ro_4 .* ro_5) * gm_4;
ro_ota = (upper_casc_ro .* lower_casc_ro) ./ (upper_casc_ro + lower_casc_ro);


% Plotting Gain vs L
plot(L, ro_ota/1e6);
xlabel('L (Channel Length) [um]','fontsize',12,'interpreter','latex');
ylabel('Output impedance M$\Omega$', 'fontsize',12,'interpreter','latex');
title('Output Impedance vs L', 'fontsize',12,'interpreter','latex');


grid on;

load 40nch_svt.mat
load 40pch_svt.mat

ro_2_OUT = 1 / (gm_2 / look_up(nch, 'GM_GDS', 'GM_ID', GM_ID_2, 'VDS', 0.2, 'L', 4))
ro_3_OUT = 1 / (gm_3 / look_up(nch, 'GM_GDS', 'GM_ID', GM_ID_3, 'VDS', 0.35, 'L', 4))
ro_4_OUT = 1 / (gm_4 / look_up(pch, 'GM_GDS', 'GM_ID', GM_ID_4, 'VDS', 0.35, 'L', 4))
ro_5_OUT = 1 / (gm_5 / look_up(pch, 'GM_GDS', 'GM_ID', GM_ID_5, 'VDS', 0.2, 'L', 4))
% Calculate parallel resistance for the specific length
parallel_ro2_ro5_OUT = (ro_2_OUT * ro_in) / (ro_2_OUT + ro_in);

% Assuming gm_3 and gm_4 are also given or calculated already
lower_casc_ro_OUT = (parallel_ro2_ro5_OUT * ro_3_OUT) * gm_3
upper_casc_ro_OUT = (ro_4_OUT * ro_5_OUT) * gm_4
ro_ota_OUT = (upper_casc_ro_OUT * lower_casc_ro_OUT) / (upper_casc_ro_OUT + lower_casc_ro_OUT)
gain_ota= ro_ota_OUT*gm_1

%Flicker noise corner (It is important to verify that the corner frequency is low enough so a lower chopping frequency can be chosen). Therefore, it is important to verify which are the noise corners for the main noise contributors (input transistors, bottom current sources, and top current
%sources).

%Bottom current tail transistor
L_bot=nch.L;
fco_bottom = look_up(nch, 'SFL_STH', 'GM_ID', 15, 'L', L_bot);

%Top current tail transistor
L_top=pch.L;

fco_top = look_up(pch, 'SFL_STH', 'GM_ID', 15, 'L', L_top);

```

```
%Input transistors

load 40pch_25.mat
L_INPUT=pch.L;

fco_input_comp = look_up(nch, 'SFL_STH', 'GM_ID', 25, 'L', L_INPUT);

% Plot fco_bottom vs L
plot(L, fco_bottom/1000, 'LineWidth', 2);
hold on; % Hold the plot so that subsequent plots are overlaid

% Plot fco_top vs L
plot(L, fco_top/1000, 'LineWidth', 2);

xlabel('Length of tail current source [um]', 'fontsize',12,'interpreter','latex');
ylabel('Frequency [kHz]','fontsize',12,'interpreter','latex');
title('Flicker noise corner frequency of top and bottom current sources','fontsize',14,'interpreter','latex');
legend('Nmos current source', '', 'Location', 'best','fontsize',12,'interpreter','latex');
grid on;

% Set x-axis limits
xlim([1, 5]);

% Create a new figure for fco_input vs L_INPUT
figure;
plot(L_INPUT, fco_input_comp/1000, 'LineWidth', 2);
xlabel('Length of tail current source (um)','fontsize',12,'interpreter','latex');
ylabel('Frequency (kHz)', 'fontsize',12,'interpreter','latex');
title('Flicker noise corner frequency of input current source', 'fontsize',14,'interpreter','latex');
grid on;


hold off; % Release the hold to prevent further plots from being overlaid

%To get the definitive sizes-
load 40nch_svt.mat
load 40pch_svt.mat

ID_W2 = look_up(nch, 'ID_W', 'GM_ID', 15, 'L', 4, 'VDS', 0.2);
ID_W3 = look_up(nch, 'ID_W', 'GM_ID', 20, 'L', 4, 'VDS', 0.35);
ID_W4 = look_up(pch, 'ID_W', 'GM_ID', 20, 'L', 4, 'VDS', 0.35);
ID_W5 = look_up(pch, 'ID_W', 'GM_ID', 15, 'L', 4, 'VDS', 0.2);

W2 = (Current_input+Current_output) / ID_W2
W3 = Current_output/ID_W3
W4 = Current_output/ ID_W4
W5 = Current_output/ ID_W5
```

A.2. General thesis calculations

This code is used to verify the transfer function, timing calculations, and noise analysis of the AFE.

```
DEFINITION OF PARAMETERS
%Constants
Kb = 1.38e-23; % Boltzmann constant
T = 300; % Temperature in Kelvin
gamma = 1; % Approximation to have some margin
BW = 500; % Signal bandwidth in Hz

1) Finding the gm
Considering:
The targe is to have approximately below 2 uVrms
1/f noise will be eliminated by chopping
The major noise contribution is from the ota, however some margin should be left from the following stages.
%INPUT BRANCH
Kb=1.38E-23;
T=300;
BW=500;
input_biasing_current=0.8e-6;
gamma=1; %To have some margin)
Current_input=input_biasing_current/2;
Current_output=0.1*input_biasing_current; %5 times smaller than the input biasing current
GM_ID_1=25; %The input is set to weak inversion
GM_ID_2=15; %moderate inversion to reduce the gm
gm_1=GM_ID_1*Current_input %This is the gm of the input transistors of the ota
gm_2=GM_ID_2*(Current_input+Current_output)
%Input-referred noise
Vint=sqrt(8*Kb*T*((gamma/gm_1)+(gamma*(gm_2/gm_1^2)))*BW)



2) Finding the reuired IBIAS
Considering:
The input transistors are biased with a gm/ID of 25.
fprintf('2.1) The required bias current for the input branch is (nA): %d\n', input_biasing_current*1e9);
fprintf('2.2) The required bias current for the output branch is (nA): %f\n', Current_output*1e9);

3) Calculation of the required values for the boxcar sampling and SCF
Considering:
That the integrating time should correspond to above 98% of the sampling time to try to push the -3dB freq
The boxcar should act as a first Low-pass filter in the transfer function
The gain should be controlled
The required capacitor ratio should be obtained
The SC-CMFB loads the output
Defining variables
Cint_boxcar_chosen=4E-12; %Integration capacitor of the boxcar
Cap_SC_cmfb=50E-15
```

```
Cint_boxcar=Cint_boxcar_chosen+Cap_SC_cmfb;

N_OSR_boxcar=4; %Setting the oversample ratio
OSR_Boxcar=2^N_OSR_boxcar; %To make sure the OSR is a power of 2.

F_Sampling_boxcar=2*BW*OSR_Boxcar;  %The required sampling frequency considering nyquist theorem
T_sampling_boxcar=1/F_Sampling_boxcar;
fprintf('3.1) The sampling frequency of the system is (KHz): %e\n', F_Sampling_boxcar/1000);
fprintf('3.2) The sampling period of the system is (us): %e\n', (T_sampling_boxcar*1e6));

Setting the Low pass filter pole:
%The -3dB pole set by the sinc response of the boxcar sampling alone is:
%Allocating the timings for the reset
Tint_Boxcar=(255/265)*T_sampling_boxcar %Allocated time period for integration of the boxcar.

TR_Boxcar=(3/265)*T_sampling_boxcar %Allocated time for the reset of the boxcar.
Ts_SC_LPD=(7/265)*T_sampling_boxcar %Allocated time period for the SCF


%Corresponding transfer function:
%Variables for the gm cell:
syms f
ro_ota=4.2e6;
w = 2*pi*f;
Req_SC_LPF = (Tint_Boxcar+Ts_SC_LPD)/Cint_boxcar;

To set the second order low pass filter pole to 500 Hz:
Cratio = solveCratio(500, Tint_Boxcar, Ts_SC_LPD);
C_SCF=Cratio*Cint_boxcar;
fprintf('3.4) The required CSCF/Cint ratio is: %e\n', Cratio);
fprintf('3.5) The required CSCF is (fF): %e\n', C_SCF*1e15);


%However, this value of capacitor can be modified, to obtained a better performance or reduce the area:

C_SCF_Chosen=0.5e-12;
Attenuation_boxcar_real=Cint_boxcar/(Cint_boxcar+C_SCF_Chosen);


fprintf('3.6) The used CSCF is (fF): %e\n', C_SCF_Chosen*1e15);
fprintf('3.7) The resulting attenuation of the boxcar sampling is (considering the choosen C_SCF): %d\n', Attenuation_boxcar_real);
To show the effect on oversampling and averaging
% Define the symbolic variable
syms f;

% Define the expressions symbolically
w = 2*pi*f;
```

```matlab
numerator = sin(pi*f*OSR_Boxcar/F_Sampling_boxcar);
denominator = OSR_Boxcar*sin(pi*f/F_Sampling_boxcar);

OSR_DECIM_LPF=abs(numerator/denominator);

% Create a function handle for the frequency response
frequency_response = matlabFunction(OSR_DECIM_LPF, 'Vars', f);

% Define a numeric frequency range for positive frequencies only
f_values = linspace(0, F_Sampling_boxcar/2, 500);  % Only positive frequencies from 0 to Fs/2

% Evaluate the function numerically over the frequency range
response_values = frequency_response(f_values);

%Gain of the Boxcar
gm_ota=10.11e-6; %The gm of the ota is the gm of the input transistors;

%To calculate the output impedance of the ota we have to use the following parameters (due to the high output impedance of the folded cascode):
%They are derived from the simulations.

ro_cascode_low=35.75e6;
ro_current_source_low=5.646e6;
ro_input=57.3e6;
ro_cascode_top=51.6e6;
ro_current_source_top=20.89e6;


gm_cascode_low=1.654e-6;
gmb_cascode_low=200e-9;

gm_cascode_top=1.62e-6;
gmb_cascode_top=210.8e-9;


%the output impedance can be approximated like this:
parallel_ro2_ro5=((ro_input*ro_current_source_low)/((ro_input+ro_current_source_low)));
ro_ota_low=(gm_cascode_low+gmb_cascode_low)*ro_cascode_low*parallel_ro2_ro5
ro_ota_top=(gm_cascode_top+gmb_cascode_top)*ro_cascode_top*ro_cascode_top
ro_ota=(ro_ota_low*ro_ota_top)/(ro_ota_low+ro_ota_top)


%Therefore the gain of the boxcar sampling would then be:

Gain_ota_amplifier_boxcar = gm_ota .* ro_ota .* (1 - exp(-Tint_Boxcar ./ (ro_ota .* Cint_boxcar)));
fprintf('3.9) The boxcar gain is (V/V): %e\n', Gain_ota_amplifier_boxcar);
fprintf('3.10) The boxcar gain after attenuation is (V/V): %e\n', Gain_ota_amplifier_boxcar*Attenuation_boxcar_real);


```

```matlab
%Now we can plot the real gain of the amplifier vs the ideal gain of an
%amplifier

t = linspace(0, Tint_Boxcar, 1000); % Time vector from 0 to tint

% Compute gains
gain_ideal_plot = gm_ota.*(t)./Cint_boxcar; % Ideal gain, constant since tint is constant
gain_real_plot = gm_ota .* ro_ota .* (1 - exp(-(t) ./ (ro_ota .* Cint_boxcar))); % Real gain, varies with time

% Plot
figure;
plot(t*1e6, gain_ideal_plot.*ones(size(t)), 'b-', 'LineWidth', 2); % Ideal gain as a horizontal line
hold on;
plot(t*1e6, gain_real_plot, 'r--', 'LineWidth', 2); % Real gain as a function of time
hold off;

title('Gain vs. Time');
xlabel('Time (us)');
ylabel('Gain');
legend('Ideal Gain', 'Real Gain', 'Location', 'Best');
grid on;


% Compute maximum values
max_ideal_gain = gm_ota*Tint_Boxcar/Cint_boxcar
max_real_gain = gm_ota * ro_ota * (1 - exp(-Tint_Boxcar / (ro_ota * Cint_boxcar)))
gain_error=((max_real_gain-max_ideal_gain)/max_ideal_gain)*100


Defining variables
Cint_boxcar_chosen=4E-12; %Integration capacitor of the boxcar
CLPF_chosen=0.5E-12; %Integration capacitor of the boxcar
Cint_boxcar=Cint_boxcar_chosen
Tint_Boxcar=61.25e-6;
Ts_SC_LPD=0.625e-6;
Req_SC_LPF = (Tint_Boxcar+Ts_SC_LPD)/Cint_boxcar
Setting the Low pass filter pole:
%Corresponding transfer function:
%Variables for the gm cell:
syms f
Req_SC_LPF = (Tint_Boxcar+Ts_SC_LPD)/Cint_boxcar;

To set the second order low pass filter pole to 500 Hz:
Cratio = solveCratio(500, Tint_Boxcar, Ts_SC_LPD)
%Also important to consider that since there is two capacitors (integrating capacitor and LPF capacitor), it will lead to attenuation. This
%attenuation is the follow:

Attenuation_boxcar=Cint_boxcar/(Cint_boxcar+CLPF_chosen);
```

```matlab
fprintf('3.6) The resulting attenuation of the boxcar sampling is (considering the calculated C_SCF) : %d\n', Attenuation_boxcar);

%However, this value of the capacitor can be modified, to obtained a better performance or reduce the area:

C_SCF_Chosen=0.5e-12;
To show the effect on oversampling and averaging
% Parameters
OSR = 16;               % Oversampling ratio
fs = 16000;               % Sampling frequency
frequencies = linspace(0, 1000e3, 10000000);  % 1000 points from 0 to 100 kHz
w = 2 * pi * frequencies;

f_p = 1/(2*pi*Req_SC_LPF*C_SCF_Chosen); % Pole frequency in Hz
w_p = 2 * pi * f_p*2; % Angular frequency of the pole

% Calculate ideal gain
Gain_ota_amplifier_boxcar = gm * Tint_Boxcar / Cint_boxcar

% Calculate LPF magnitude response
LPF_magnitude = 1 ./ sqrt(1 + (w ./ w_p).^2); %This is what the pole doesn't match

%Additional pole from the OTA

% Calculate boxcar filter magnitude response
boxcar_magnitude = abs(Gain_ota_amplifier_boxcar) .* abs((sin(pi.*frequencies.*(0.99*(1/fs)))./(pi.*frequencies*(0.99*(1/fs)))));
SCF_SINC = abs(1) .* abs((sin(pi.*frequencies.*(0.99*(1/fs)))./(pi.*frequencies*(0.99*(1/fs)))));

H_formula = (1/OSR) .* abs(sin(pi .* frequencies .* OSR ./ fs) ./ (pi .* frequencies ./ fs));
%Include accumulating part

% Combined transfer function magnitudes
combined_magnitude = SCF_SINC .* boxcar_magnitude.*LPF_magnitude;
combined_magnitude_OS = combined_magnitude.*H_formula;

% Convert magnitudes to dB scale
combined_dB = 20 * log10(combined_magnitude);
combined_dB_OS = 20 * log10(combined_magnitude_OS);
LPF_dB = 20 * log10(LPF_magnitude);
boxcar_dB = 20 * log10(boxcar_magnitude);
H_formula_dB = 20 * log10(H_formula);

% Handling -Inf/NaN in dB conversions
combined_dB(isinf(combined_dB) | isnan(combined_dB)) = -100;
combined_dB_OS(isinf(combined_dB_OS) | isnan(combined_dB_OS)) = -100;
LPF_dB(isinf(LPF_dB) | isnan(LPF_dB)) = -100;
boxcar_dB(isinf(boxcar_dB) | isnan(boxcar_dB)) = -100;

% Plot LPF magnitude response
```

```
figure;
semilogx(frequencies, LPF_dB, 'LineWidth', 2);
title('SCF (Switched Capacitor Filter) Frequency Response in dB Scale');
xlabel('Frequency (Hz)');
ylabel('Gain [dB]');
grid on;


% Plot boxcar filter magnitude response
figure;
semilogx(frequencies, boxcar_dB, 'LineWidth', 2);
title('Boxcar Filter Frequency Response in dB Scale');
xlabel('Frequency [Hz]');
ylabel('Gain [dB]');
grid on;
xlim([10, 100000]);
ylim([-120, 100]);
yticks(-120:20:100)
% Plot combined transfer function (Boxcar + SCF)
figure;
semilogx(frequencies, combined_dB, 'LineWidth', 2);
title('System Frequency Response (Boxcar + SCF) in dB Scale');
xlabel('Frequency [Hz]');
ylabel('Gain [dB]');
grid on;
xlim([10, 100000]);
ylim([-120, 100]);
yticks(-120:20:100)

% Plot combined transfer function with OSR (Boxcar + SCF + Decimator)
figure;
semilogx(frequencies, combined_dB_OS, 'LineWidth', 2);
title('System Frequency Response (Boxcar + SCF + Accumulator) in dB Scale');
xlabel('Frequency (Hz)');
ylabel('Gain [dB]');
grid on;
xlim([10, 100000]);
ylim([-120, 100]);
% Plot the magnitude response from the given formula
figure;
plot(frequencies, H_formula_dB, 'LineWidth', 2);
title('Magnitude Response of the Given Formula');
xlabel('Frequency (Hz)');
ylabel('Magnitude (dB)');
grid on;
xlim([0, max(frequencies)]);
ylim([-120, 100]);
yticks(-120:20:100)
xlim([0, 100000]);
% Find -3 dB cutoff frequency for combined response without OS
combined_magnitude_normalized = combined_magnitude / max(combined_magnitude);
[~, index] = min(abs(combined_magnitude_normalized - 0.7071));
f_3dB = frequencies(index);  % Frequency at -3 dB point
```

```
disp(['-3 dB cutoff frequency: ', num2str(f_3dB), ' Hz']);

% Find -3 dB cutoff frequency for combined response with OS
combined_magnitude_normalized_OS = combined_magnitude_OS / max(combined_magnitude_OS);
[~, index_OS] = min(abs(combined_magnitude_normalized_OS - 0.7071));
f_3dB_OS = frequencies(index_OS);  % Frequency at -3 dB point
disp(['-3 dB cutoff frequency with OS: ', num2str(f_3dB_OS), ' Hz']);

5) Continuous time comparator (ADC)
VDD= 1.1; %Supply of the comparator

%Input ranges (Peak)
Input_swing=2e-3;
Max_LFP_input=0.5e-3; %this is the input LFP amplitude
Max_IR_EDO=Input_swing-Max_LFP_input; %this is only Vpeak of the IR_offset after compensation
Input_range_comparator=Gain_ota_amplifier_boxcar*Input_swing;
fprintf('5.1) The input range of the continuous-time comparator is Vp (mV): %e\n', Input_range_comparator*1000);
%Number of Codes
N_ramp=8; %Number of bits of the LSB
Number_of_codes=2^N_ramp; %To calculate the total output of codes

Ratio_EDO_LFP=Max_LFP_input/(Max_LFP_input+Max_IR_EDO); %To check which is the relationship between the input signal and the input offset

%Number of LFP Bits
Useful_codes=Number_of_codes*Ratio_EDO_LFP; %To check which are the codes that are set to the LFP signal
n_lfp = log2(Useful_codes); %To see how many bits are actually used for the signal or lfp
fprintf('5.2) The Number of bits used for the LFP are actually %f \n',n_lfp);

Extra_bit_oversampled=OSR_Boxcar/4;
Adc_useful_bits_oversampled=n_lfp+Extra_bit_oversampled;
fprintf('5.3) The Number of bits used for the LFP after oversampling are actually %f \n',Adc_useful_bits_oversampled);

%To check the required frequency
Required_Tclk=2^N_ramp;

Comp_latency=2; %Periods to allow for comparator latency
Total_need_Tclk=Required_Tclk+Comp_latency+7; %Also including the treset time

%The number of allocated periods
Tcomp_Tclk=Total_need_Tclk; %The integration time is set to be the same as the ramp ADC needs to make comparisons
Required_CLK_freq=Total_need_Tclk*F_Sampling_boxcar;
fprintf('5.4) The required main CLK frequency is (MHz)%d \n',Required_CLK_freq/1e6);

fprintf('5.5) The required main CLK period is (us)%d \n',(1/Required_CLK_freq)*1e6
```

```
    );
fprintf('5.6) The required Tramp period without lattency (us)%d \n',Required_Tclk
    *(1/Required_CLK_freq)*1e6);

%The minimum SR of the continuous time comparator can be determined as:
SR_CC_min= Required_Tclk*VDD/(Required_Tclk*(1/Required_CLK_freq));
fprintf('5.7) The required SR for the comparator is (V/us)%d \n',SR_CC_min/1e6);


%6) Chopping
%To consider:
%fchop >  BW + 1/f noise corner
The higher the chopping frequency the more attenuation will there be as the input
    impedance drecreases proportional to fchop. The chopping frequency has to be a
    multiple of the sinc function to get rid of the HF chopping ripple provinient
    from there.

N_chop_multiple=3; %How many multiples of the sinc function are selected for the
    chopping frequency.
F_chop_used_tsamp= 1/T_sampling_boxcar*N_chop_multiple; %This is the chopping
    frequency used for the system if we consider the notch as multiple of 1/tsamp.
F_chop_used_tint= 1/Tint_Boxcar*N_chop_multiple; %This is the chopping frequency
    used for the system if we consider the notch as multiple of 1/tint.


%7) Timing diagram
%To consider:
%The tramp, tint, trst, tlpf
%The chopping periods
%The ramp signal
%The comparison clock signal
%The clear signals
%First, we can check how many periods are needed for tramp, tint, trst, tlpf:
fprintf('7.1) The Number of clock periods for the comparison time %d \n',
    Tcomp_Tclk);

%The time needed for the integration is the time needed for the ramp - the
%time needed for the reset

TR_tclk=3; %If we assign 2 periods for the reset
TS_tclk=7; %If we assign 2 periods for the reset
T_int=Tcomp_Tclk-TR_tclk-TS_tclk; %We have the time for the boxcar integration


fprintf('7.2) The Number of clock periods for the SCF  %d \n',TS_tclk);
fprintf('7.3) The Number of clock periods for the reset %d \n',TR_tclk);
fprintf('7.5) The Number of clock periods for the INT %d \n',T_int);
fprintf('7.6) The Number of clock periods for the an entire period %d \n',
    Tcomp_Tclk);


%Now the number of clock periods allocated for the chopping:
T_chop_tint=(Required_CLK_freq/F_chop_used_tint); %It should be multiple of 2 to
```

```
make the DC easier!
T_chop=(Required_CLK_freq/F_chop_used_tsamp)
fprintf('7.7) The Number of clock periods for the chopping %d \n', 86); %It should be multiple of 2 to make the DC easier!
fprintf('7.8) The chopping frequency used is (KHz) %d \n',Required_CLK_freq/86);

AUXILIAR FUNCTIONS
Function to obtain the ratio of CSCF/Cint (Check if I need the pi in the sinc equation).
function Cratio = solveCratio(f, tint, ts)
    % Objective function to be solved
    func = @(Cratio) abs(1/sqrt(1 + (2*pi*f*Cratio*(tint+ts))^2))*abs(sin(f*tint)/(f*tint)) - (1/sqrt(2));

    % Initial guess for Cratio
    Cratio_initial_guess = 0.3; % Adjust based on your expected range of Cratio

    % Solve for Cratio
    options = optimoptions('fsolve', 'Display', 'none', 'FunctionTolerance', 1e-14);
    Cratio = fsolve(func, Cratio_initial_guess, options);
end
```

A.3. Signal analysis

This code is used to obtain the frequency spectrum analysis and the signal reconstruction.

```
Frequency Analysis and signal reconstruction
VREF=320 mV, Input signal: 1 mVpp
% Define the file paths
no_noise_file = 'C:\Users\arnau\Desktop\thesis\MATLAB_THESIS\Bulk_mod_folded_casc\FINAL\FFT_spectrum_VREF_320mV_no_noise.csv';
noise_file = 'C:\Users\arnau\Desktop\thesis\MATLAB_THESIS\Bulk_mod_folded_casc\FINAL\FFT_spectrum_VREF_320mV_noise.csv';

% Load the data
no_noise_data = readtable(no_noise_file);
noise_data = readtable(noise_file);

% Extract frequency and magnitude
freq_no_noise = no_noise_data{:, 1};
mag_no_noise = no_noise_data{:, 2};

freq_noise = noise_data{:, 1};
mag_noise = noise_data{:, 2};

% Convert magnitude to dBFS
max_mag_no_noise = max(mag_no_noise);
max_mag_noise = max(mag_noise);

mag_no_noise_dbfs = mag_no_noise - max_mag_no_noise;
mag_noise_dbfs = mag_noise - max_mag_noise;
```

```

% Plot the data

% Plot for No Noise
figure;
for k = 1:length(freq_no_noise)
    line([freq_no_noise(k), freq_no_noise(k)], [-250, mag_no_noise_dbfs(k)], 'Color', 'b', 'linewidth', 1.5);
end
title('AFE output spectrum (No Noise)');
xlabel('Frequency [Hz]');
ylabel('Magnitude [dBFS]');
ylim([-200, 0]);
grid on;
set(gca, 'LineWidth', 1.5, 'TickDir', 'in', 'TickLength', [0.01, 0.01]);
set(gca, 'Box', 'on');

% Plot for Noise
figure;
for k = 1:length(freq_noise)
    line([freq_noise(k), freq_noise(k)], [-250, mag_noise_dbfs(k)], 'Color', 'r','linewidth', 1.5);
end
title('AFE output spectrum (With Noise)');
xlabel('Frequency [Hz]');
ylabel('Magnitude [dBFS]');
ylim([-100, 0]);
grid on;
set(gca, 'LineWidth', 1.5, 'TickDir', 'in', 'TickLength', [0.01, 0.01]);
set(gca, 'Box', 'on');
STATS FROM THE SIGNAL WITHOUT NOISE
% Calculate SNDR, SFDR, THD, and ENOB for the signal without noise
% Display the results for the signal without noise
fprintf('Analysis for the signal without noise:\n');
fprintf('Fundamental Frequency: 100 Hz\n');
fprintf('SNDR: %.2f dB\n', 51.73);
fprintf('SFDR: %.2f dBc\n', 54);
fprintf('ENOB: %.2f bits\n', 8.3);

STATS FROM THE SIGNAL WITH NOISE
% Assuming freq_noise and mag_noise are loaded and represent frequency and magnitude in dBFS

% Normalize amplitude to dBFS
mag_noise_dBFS = mag_noise - max(mag_noise);

% Define the bandwidth of interest (500 Hz)
bandwidth_of_interest = 500;

% Find indices within the specified bandwidth
bandwidth_indices = freq_noise <= bandwidth_of_interest;

% Calculate the noise power within the specified bandwidth
```

```matlab
% Convert dBFS to linear scale before summing
noise_power_linear = sum(10.^(mag_noise_dBFS(bandwidth_indices) / 10));

% Calculate the integrated noise power in dB
integrated_noise_power_dB = 10 * log10(noise_power_linear);

% Input signal peak-to-peak voltage (500 uVpp)
V_pp = 500e-6;

% Convert peak-to-peak voltage to RMS voltage
V_RMS = V_pp / (2 * sqrt(2));

% Calculate the integrated noise voltage (RMS)
% Convert the integrated noise power back to voltage
integrated_noise_voltage_RMS = sqrt(noise_power_linear) * V_RMS;

% Convert integrated noise voltage to uV
integrated_noise_voltage_uV = integrated_noise_voltage_RMS * 1e6;

% Define the gain
gain = 145;

% Find the fundamental frequency at 100 Hz
fundamental_idx = find(freq_noise == 100);
if isempty(fundamental_idx)
    error('No signal at 100 Hz found in the data.');
end
fundamental_mag = mag_noise(fundamental_idx);

% Calculate total power of the fundamental frequency
signal_power = 10^(fundamental_mag / 10);

% Exclude DC and the fundamental frequency when calculating noise and harmonics
noise_indices = find(freq_noise ~= 0 & freq_noise ~= 100); % Excludes DC component (0 Hz)
noise_powers = 10.^(mag_noise(noise_indices) / 10);
total_noise_power = sum(noise_powers);

% Calculate SNR (Signal to Noise Ratio)
SNR = 10 * log10(signal_power / total_noise_power);

% Calculate harmonics (excluding the fundamental)
harmonic_indices = find(mod(freq_noise, 100) == 0 & freq_noise ~= 100 & freq_noise ~= 0); % also excludes DC
harmonic_powers = 10.^(mag_noise(harmonic_indices) / 10);
total_harmonic_power = sum(harmonic_powers);

% Calculate SNDR (Signal to Noise and Distortion Ratio)
total_distortion_power = total_harmonic_power + total_noise_power;
SNDR = 10 * log10(signal_power / total_distortion_power);

% Calculate SFDR (Spurious-Free Dynamic Range)
spurious_powers = noise_powers;
```

```
[max_spur_power, ~] = max(spurious_powers);
SFDR = 10 * log10(signal_power / max_spur_power);

% Calculate ENOB (Effective Number of Bits)
ENOB = (SNDR - 1.76) / 6.02;

% Calculate THD (Total Harmonic Distortion) in dB
THD_ratio = sqrt(total_harmonic_power / signal_power);
THD = 20 * log10(THD_ratio); % in dB

% Display the results
fprintf('Fundamental Frequency: 100 Hz\n');
fprintf('Amplitude: 1 mVpp');
fprintf('SNR: %.2f dB\n', SNR);
fprintf('SNDR: %.2f dB\n', SNDR);
fprintf('SFDR: %.2f dBc\n', SFDR);
fprintf('ENOB: %.2f bits\n', ENOB);
fprintf('THD: %.2f dB\n', THD);
fprintf('Integrated Noise Voltage (RMS): %.2f uV\n', integrated_noise_voltage_uV / gain);

RECONSTRUCTION
OUTPUT FROM THE AFE (Vin=1 mVpp at 100 Hz)
linear_gain = 148.34;

% Load the data from the CSV file
data = readtable('C:\Users\arnau\Desktop\thesis\MATLAB_THESIS\Bulk_mod_folded_casc\FINAL\FFT_TRAN_VREF_320_mV_noise.csv');

% Extract the columns by index
time = data{:,1}; % First column
signal = data{:,2}; % Second column

% --- Plot Original Signal ---
figure;
plot((time-23e-3)*1000, 1000*signal, 'LineWidth', 2);
xlabel('Time [ms]', 'FontSize', 12, 'FontWeight', 'bold');
ylabel('Amplitude [mV]', 'FontSize', 12, 'FontWeight', 'bold');
title('AFE output', 'FontSize', 14, 'FontWeight', 'bold');
grid on;
yticks1 = linspace(-100, 100, 11);  % Add more y-ticks for the first graph
set(gca, 'YTick', yticks1, 'XTick', 0:5:40, 'LineWidth', 1.5, 'TickDir', 'in', 'TickLength', [0.02, 0.02], 'FontSize', 12, 'FontWeight', 'bold');
ylim([-100 100]);

% Set the x-axis limits
xlim([0 40]); % 23 ms to 63 ms

% --- INTERPOLATION OF OUTPUT SIGNAL ---
% Find the data point closest to 23 ms
[~, closest_index] = min(abs(time - 23e-3));

% Extract the signal value and time at the closest point
```

```matlab
signal_peak = signal(closest_index);
time_peak = time(closest_index);

% Define sine wave parameters (adjusted for peak at 23 ms)
f = 100; % Frequency (Hz)
amp = 0.5e-3; % Amplitude (1 mVpp / 2 for peak-to-peak)
phase_shift = 2*pi*f*(time_peak - 23.55e-3);  % Introduce phase shift to make the signals in sinc

% Generate a finer time vector for the sine wave
t_sine_fine = linspace(min(time), max(time), length(time)*10);

% Perform spline interpolation on the signal
signal_interp = spline(time, signal, t_sine_fine);

% Generate the sine wave with adjusted amplitude and phase shift
sine_wave_scaled = amp * sin(2*pi*f*t_sine_fine + phase_shift);

% --- Plot Spline Interpolation ---
figure;
plot((t_sine_fine-23e-3)*1000, 1000*signal_interp/linear_gain, 'LineWidth', 2);
xlabel('Time [ms]', 'FontSize', 12, 'FontWeight', 'bold');
ylabel('Amplitude [mV]', 'FontSize', 12, 'FontWeight', 'bold');
title('Reconstructed AFE Output', 'FontSize', 14, 'FontWeight', 'bold');
grid on;
yticks2 = linspace(-0.6, 0.6, 13);
set(gca, 'YTick', yticks2, 'XTick', 0:5:40, 'LineWidth', 1.5, 'TickDir', 'in', 'TickLength', [0.01, 0.01], 'FontSize', 12, 'FontWeight', 'bold');

% Set the x-axis limits (adjust y-axis if needed based on interpolation range)
xlim([0 40]);

% --- Plot Original Signal and Sine Wave ---
figure;
plot((t_sine_fine-23e-3)*1000, 1000*signal_interp/linear_gain, 'LineWidth', 2);
hold on;  % Hold the plot to add elements on the same graph
% Plot the sine wave (shifted for peak at 23 ms)
plot((t_sine_fine-23e-3)*1000, sine_wave_scaled*1000, 'r--', 'LineWidth', 2.5);
hold off;

xlabel('Time [ms]', 'FontSize', 12, 'FontWeight', 'bold');
ylabel('Amplitude [mV]', 'FontSize', 12, 'FontWeight', 'bold');
title('Reconstructed AFE Output with 1 mVpp Sine-wave input', 'FontSize', 14, 'FontWeight', 'bold');
grid on;

% Set the x-axis limits
xlim([0 40]);
ylim([-0.8 0.8]);
yticks3 = linspace(-0.8, 0.8, 17);
set(gca, 'YTick', yticks3, 'XTick', 0:5:40, 'LineWidth', 1.5, 'TickDir', 'in', 'TickLength', [0.01, 0.01], 'FontSize', 12, 'FontWeight', 'bold');

```

```matlab
% Legend entries
legendEntries_FFT = {'Reconstructed signal ', 'Input sinewave'};
% Add legend with desired font settings
legend(legendEntries_FFT, 'Location', 'Best', 'FontSize', 10, 'FontWeight', 'bold');
```

B

Verilog-A codes

This Appendix exposes the Verilog-A codes used in the designed AFE.

B.1. On-chip clock generator

This code generates all the necessary control signals for the implemented AFE using an input high-frequency clock signal.

```
`include "constants.vams"
`include "disciplines.vams"

module ClockGenerator(
    input clk,          // Base high-frequency clock
    input reset,        // Asynchronous reset
    output Tint,        // Output clock Tint
    output Tlpf,        // Output clock Tlpf
    output Trst,        // Output clock Trst
    output Tcomp,       // Output clock Tcomp
    output Tdecim,       // Output clock Tcomp
    output Tclear_adc,  // Output clock Tclear_adc
    output Tclear_dec,  // Output clock Tclear_dec
    output Tclear_dac,  // Output clock Tclear_dac
    output Tchop,        // CHOP output
    output T_out_ready,  // The output is decimated and ready
    output Tclear_dec_out // The output register is cleared

);

// Define clk and reset as electrical types
electrical clk, reset;
electrical Tint, Tlpf, Trst, Tcomp, Tclear_adc, Tclear_dec,Tclear_dac,Tdecim,
    Tchop, T_out_ready, Tclear_dec_out;

parameter real vtrans_clk = 0.55;      // Threshold for clock transition
parameter real vlogic_high = 1.1;      // Logic high voltage level (VHIGH)
parameter real vlogic_low = 0;         // Logic low voltage level (VLOW)
parameter real trise = 10n;            // Rise time
parameter real tfall = 10n;            // Fall timeE

integer count = 0;
integer count_chop = 0;  // Separate counter for chopping frequency
integer count_decimation=0;
integer count_compare=0;
integer count_dac=0;
integer clear_dac_flag = 0;  // Flag to control clear_dac, goes high once and stays high
integer tcomp_active = 0;  // Flag to activate Tcomp after skipping one full cycle


analog begin

    @(cross(V(clk) - vtrans_clk, +1)) begin
        if (V(reset) < vtrans_clk) begin
            count = 0;  // Reset the main counter on reset signal
            count_chop = 0;  // Reset the chop counter on reset signal
                        count_decimation=0;
                        count_dac=0;
                        clear_dac_flag = 0;  // Flag to control clear_dac, goes high once and stays high
                        count_compare=0;
        end else begin
            count = (count == 264) ? 0 : count + 1;  // Increment or reset count based on period
            count_chop = (count_chop == 85) ? 0 : count_chop + 1;  // Increment or reset chop count based on chop period
                    count_decimation=(count_decimation==4239) ? 0: count_decimation + 1;
                        count_dac=count_dac+1;
                        count_compare = (count_compare == 529) ? 0 : count_compare + 1;
                if (count == 264 && !tcomp_active) begin
                tcomp_active = 1;  // Set Tcomp to be active from the next cycle onwards
        end

                if (count==264) begin
                        clear_dac_flag = 1;  // Flag to control clear_dac, goes high once and stays high
                end

        end
end
        // Generate the other clock signals with specific timing
        V(Tint) <+ transition((count >= 10 && count < 265) ? vlogic_high : vlogic_low, trise, tfall);
        V(Tlpf) <+ transition((count >= 0 && count < 7) ? vlogic_high : vlogic_low, trise, tfall);
        V(Trst) <+ transition((count >= 7 && count < 10) ? vlogic_high : vlogic_low, trise, tfall);
            V(Tdecim) <+ transition((count >= 0 && count < 3 && tcomp_active == 1) ? vlogic_high : vlogic_low, trise, tfall);
        V(Tcomp) <+ transition((count_compare >= 0 && count_compare < 3 && tcomp_active == 1) ? vlogic_high : vlogic_low, trise, tfall);

// Modify Tclear_adc, Tclear_dec, and Tclear_dac signals to start at 0 for at least 4 counts
```

```
    V(Tclear_adc) <+ transition((count >= 0 && count < 3 && tcomp_active == 0) ?
        vlogic_low : ((count >= 3 && count < 7) ? vlogic_low : vlogic_high), trise,
         tfall);

    V(Tclear_dec) <+ transition((count_decimation >= 0 && count_decimation < 3 &&
        tcomp_active == 0) ? vlogic_low : ((count_decimation >= 5 &&
        count_decimation < 7) ? vlogic_low :
    vlogic_high), trise, tfall);
    V(T_out_ready) <+ transition((count_decimation >= 3 && count_decimation < 5) ?
         vlogic_high : vlogic_low, trise, tfall);

    V(Tclear_dac) <+ transition((count >= 0 && count < 3 && clear_dac_flag == 0 &&
         tcomp_active == 0) ? vlogic_low : ((count >= 3 && count < 7 &&
        clear_dac_flag == 0) ? vlogic_low : vlogic_high), trise, tfall);

    V(Tclear_dec_out) <+ transition((count_decimation >= 0 && count_decimation < 3
         && tcomp_active == 0) ? vlogic_low : vlogic_high, trise, tfall);

    // Use the new count_chop for Tchop signal generation
    V(Tchop) <+ transition((count_chop < 43) ? vlogic_high : vlogic_low, trise,
        tfall);  // CHOP 44 periods high, 44 periods low
end
endmodule
```

C

Digital logic

This Appendix exposes the truth table and the logic gate diagram of the logic blocks implemented with Logic Friday software.

Table C.1: Truth Table of the control logic for the UP/DW register used in the BLC.

A[5]	**A[4]**	**A[3]**	**A[2]**	**A[1]**	**A[0]**	**Pol**	**UP**	**DW**	**STOP_UP**	**Pol_new**	**STOP_DW**
1	1	1	1	1	1	0	0	1	1	0	0
1	1	1	1	1	1	1	1	0	1	0	0
1	X	X	X	X	X	1	X	X	0	1	0
X	1	X	X	X	X	1	X	X	0	1	0
X	X	1	X	X	X	1	X	X	0	1	0
X	X	X	1	X	X	1	X	X	0	1	0
X	X	X	X	1	X	1	X	X	0	1	0
X	X	X	X	X	1	1	X	X	0	1	0
X	X	X	X	X	X	1	1	X	0	1	0
X	X	X	X	X	X	1	X	0	0	1	0
0	0	0	0	0	0	X	1	0	0	1	0
0	0	0	0	0	0	X	X	X	0	0	1

Table C.2: Truth Table of the two-bit adder used in the accumulating procedure of the oversampled ADC.

A1	A0	B1	B0	CIN	S1	S0	Cout
1	1	1	1	X	1	0	0
0	1	0	1	X	1	0	0
0	0	1	0	X	1	0	0
1	0	0	0	X	1	0	0
1	1	1	X	1	1	0	0
0	1	0	X	1	1	0	0
1	X	1	1	1	1	0	0
0	X	0	1	1	1	0	0
0	0	1	X	0	1	0	0
1	0	0	X	0	1	0	0
0	X	1	0	0	1	0	0
1	X	0	0	0	1	0	0
X	1	X	1	1	0	1	0
X	0	X	0	1	0	1	0
X	0	X	1	0	0	1	0
X	1	X	0	0	0	1	0
1	X	1	X	X	0	0	1
1	1	X	1	X	0	0	1
X	1	1	1	X	0	0	1
1	1	X	X	1	0	0	1
X	1	1	X	1	0	0	1
1	X	X	1	1	0	0	1
X	X	1	1	1	0	0	1

Table C.3: Truth table of the 4-bit adder used in the accumulating procedure of the oversampled ADC.

A3	A2	A1	A0	Cin	O3	O2	O1	O0
1	X	X	X	X	1	0	0	0
X	1	1	1	1	1	0	0	0
X	1	0	X	X	0	1	0	0
X	1	X	0	X	0	1	0	0
X	1	X	X	0	0	1	0	0
X	0	1	1	1	0	1	0	0
1	X	1	1	1	0	1	0	0
X	X	1	0	X	0	0	1	0
X	X	1	X	0	0	0	1	0
X	X	0	1	1	0	0	1	0
1	1	X	1	1	0	0	1	0
X	X	X	0	1	0	0	0	1
X	X	X	1	0	0	0	0	1
1	1	1	X	1	0	0	0	1

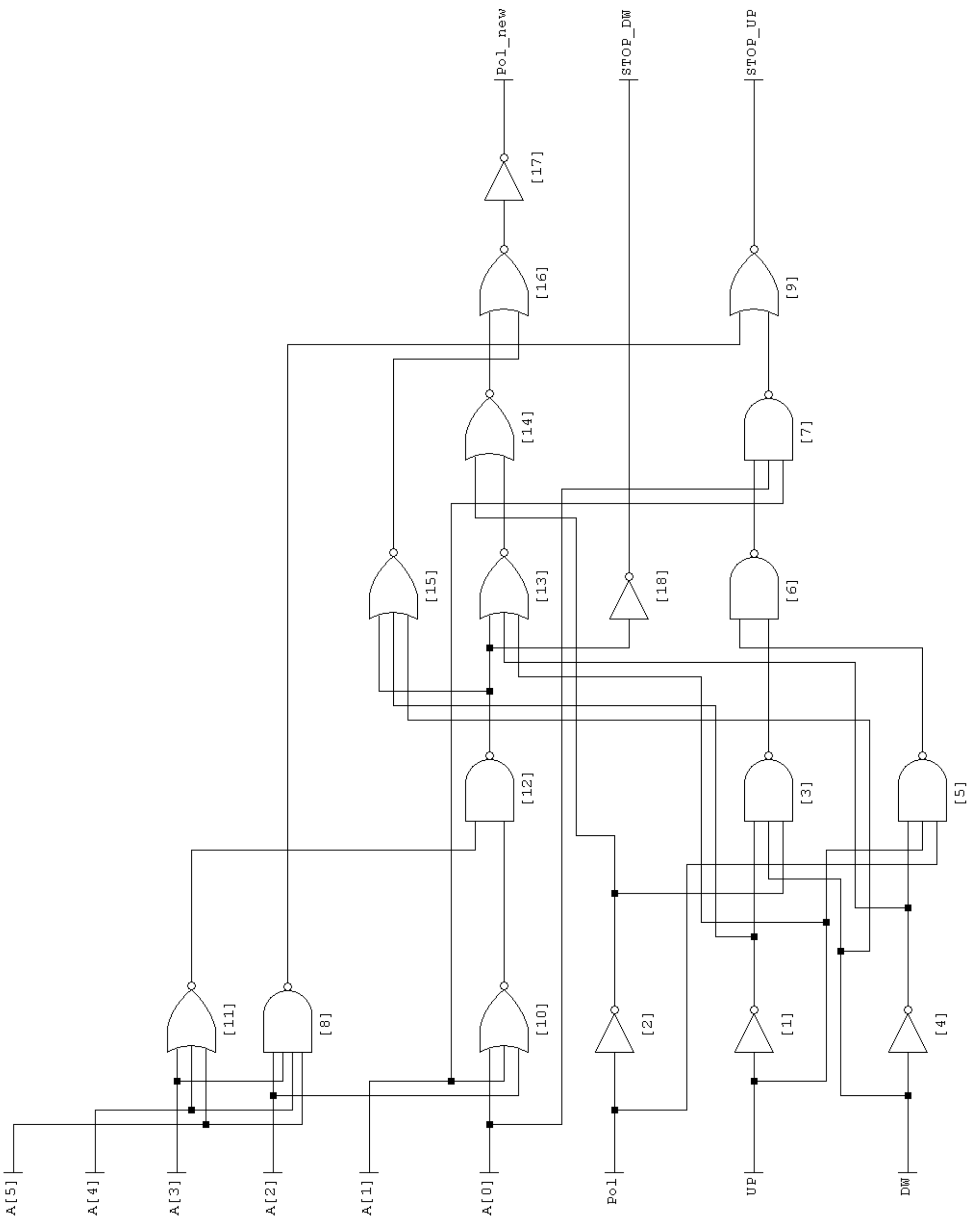


Figure C.1: Gate diagram of the control logic for the UP/DW register used in the BLC.

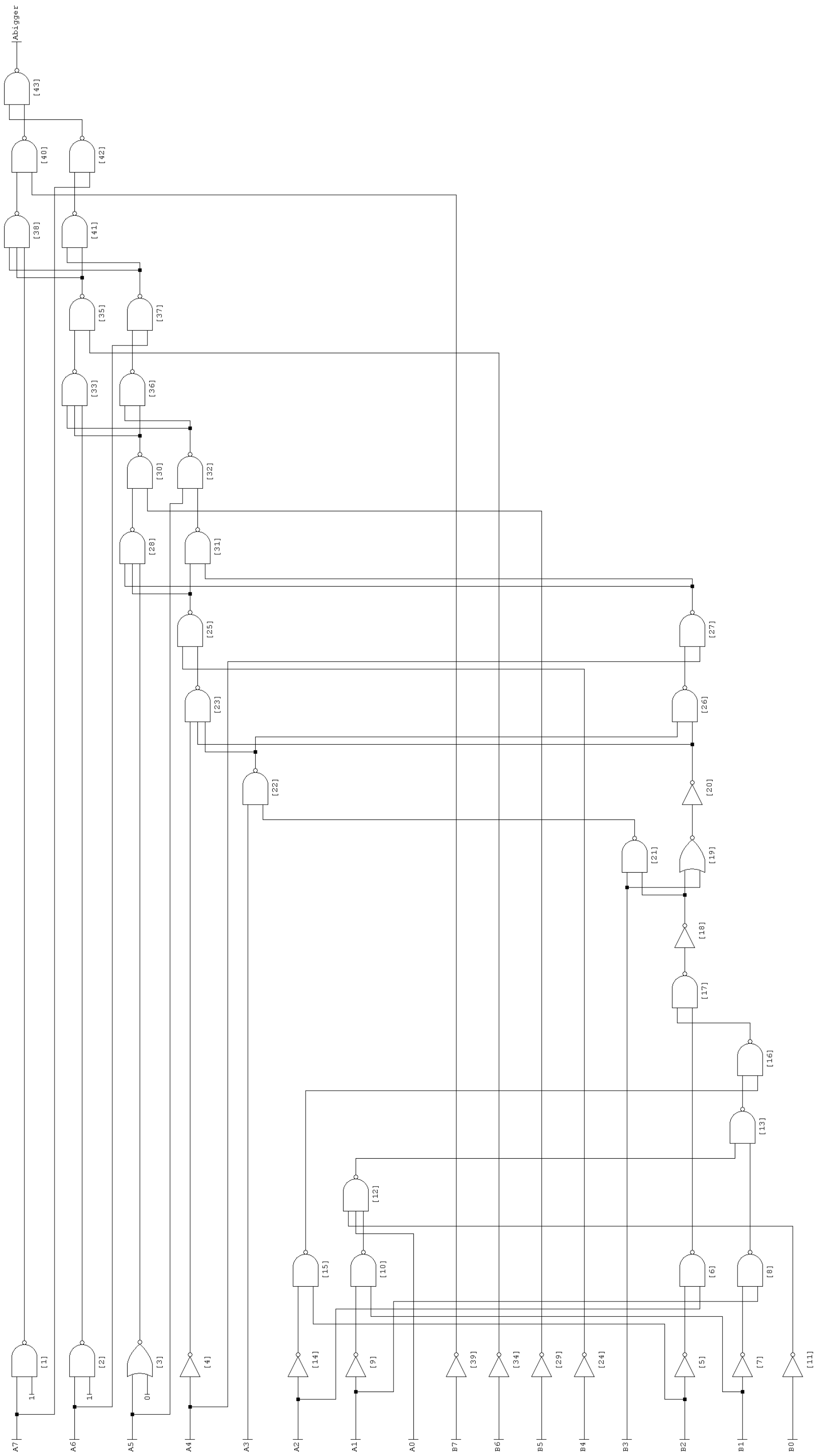


Figure C.2: Gate diagram of the digital 8-bit comparators.

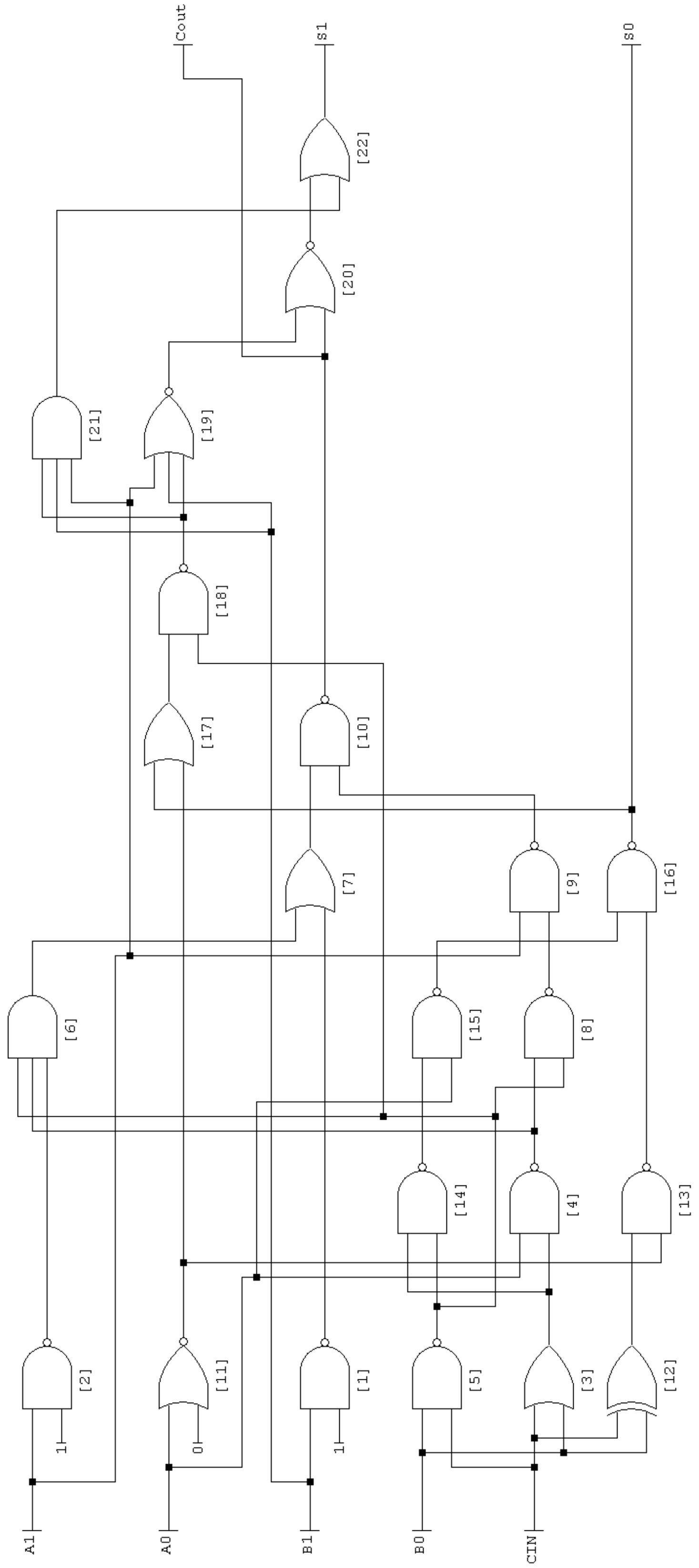


Figure C.3: Gate diagram of the two-bit adder used in the accumulating procedure of the oversampled ADC.

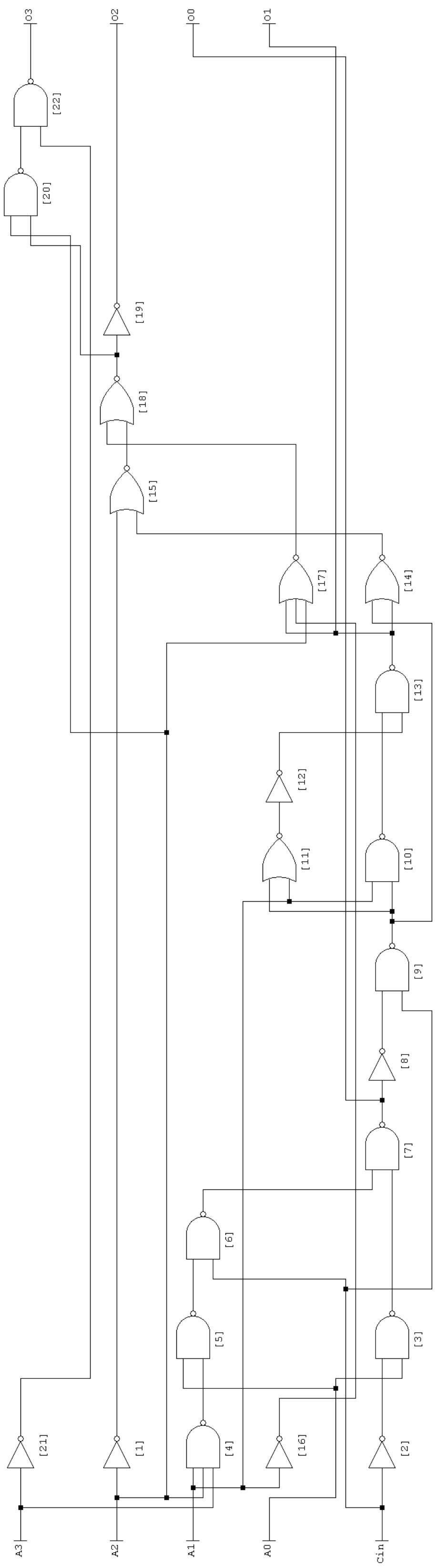


Figure C.4: Gate diagram of the 4-bit adder used in the accumulating procedure of the oversampled ADC.

D

Preliminary AFE Layout

This Appendix presents the preliminary layout with all components of the AFE placed adjacently to estimate the area of the design. This preliminary layout doesn't account for routing, spacing requirements, or other layout considerations that may increase the final area.

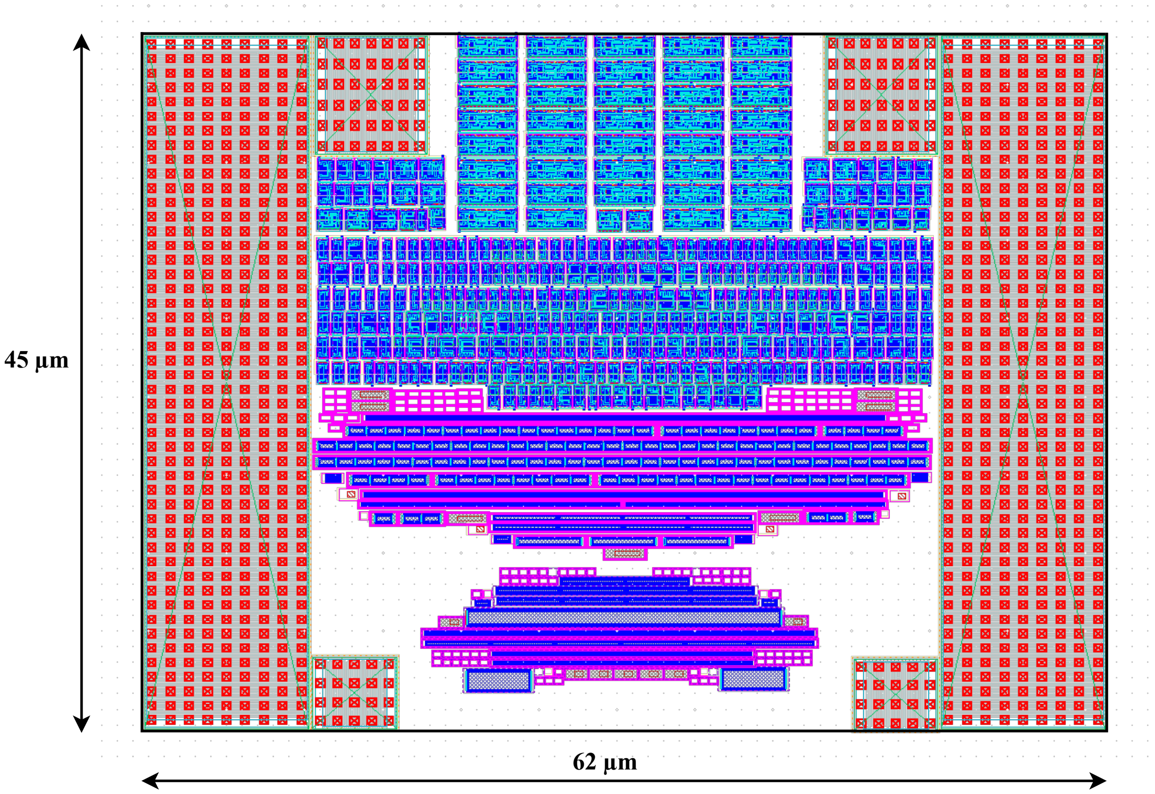


Figure D.1: Preliminary AFE layout for area estimation.