

A HIGHLY-SCALABLE DC-COUPLED DIRECT-ADC
NEURAL RECORDING CHANNEL ARCHITECTURE WITH
INPUT-ADAPTIVE RESOLUTION

SAYEDEH MINA SAYEDI

A THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

GRADUATE PROGRAMME IN COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

SEPTEMBER 2022

© SAYEDEH MINA SAYEDI 2022

ABSTRACT

This thesis presents the design, development, and characterization of a novel neural recording channel architecture with (a) quantization resolution that is adaptive to the input signal's level of activity, (b) fully-dynamic power consumption that is linearly proportional to the recording resolution, and (c) immunity to DC offset and drifts as well as artifacts at the input. Our results demonstrate the proposed design's capability in conducting neural recording with near lossless input-adaptive data compression, leading to a significant reduction in the energy required for both recording and data transmission, hence allowing for a potential high scaling of the number of recording channels integrated on a single implanted microchip without the need to increase the power budget.

The proposed design adopts a neural ADC with a novel integrating-summing feedback DAC that removes the need for area-intensive multi-bit capacitive/resistive DACs reported in state of the art, leading to a substantial reduction in the required silicon area for each channel, and more importantly, very promising design scalability with CMOS technology nodes. The proposed neural recording channel architecture is capable of removing input DC offsets and drifts as well as all other low-frequency undesired interferences such as motion/stimulation artifacts. The proposed channel with the implemented compression technique is implemented in a standard 130nm CMOS technology with overall power consumption of 7.6 μ W and active area of 92 \times 92 μ m for the implemented digital-backend.

ACKNOWLEDGMNET

First and foremost, I would like to sincerely thank my supervisor Dr. Hossein Kassiri. About two years of continuous work with Dr. Kassiri was a great and rewarding experience. I gained great insights into this field through him. He patiently directed my research and enlightened me with valuable advice and guidance to make this work possible.

I would like to express my gratitude to my committee members, Dr. Amir Sodagar, Dr. Regina Lee and Dr. Amirali Amirsoleimani for their great feedback and valuable comments on this thesis.

I would like to thank my colleagues Mahdi Nekoui and Tayebeh Yousefi who patiently shared their experience and knowledge of CAD tools with me and making this research possible.

I would also like to thank my colleagues and ICSL members, Alireza Dabbaghian, Tania Moeinfard, Mansour Taghadosi and Parsa Farshatfar for making my master's studies a memorable experience.

Finally, I want to express my deepest gratitude to my parents and my family members Sara and Mehran for always providing me with continuous encouragement and motivation throughout the years and through the process of researching and writing this thesis.

TABLE OF CONTENTS

ABSTRACT.....	ii
ACKNOWLEDGMNET.....	iii
TABLE OF CONTENTS.....	iv
LIST OF TABLES.....	viii
LIST OF FIGURES	vii
LIST OF ACRONYMS	xi
Chapter 1 Introduction and Motivation	1
1.1 Motivation and objectives	1
1.1.1 Implantable BMIs.....	3
1.1.2 Power constraints for channel count scaling	4
1.2 Data Compression	6
1.2.1 Introduction and trade-offs	6
1.2.2 Information sparsity.....	8
1.2.3 Compressive sensing	8
1.2.4 Adaptive sampling.....	10
1.3 Adaptive resolution	12

1.4 Thesis Organization.....	14
Chapter 2 System-Level Implementation.....	15
2.1 Background Theory.....	16
2.1.1 Events in Neural Recording.....	16
2.1.2 Neural ADCs	17
2.1.3 Adaptive Resolution vs. Adaptive Sampling.....	19
2.2 Functional Implementation	23
2.2.1 Digital Back-End	23
2.2.2 Mixed-Signal Front-End.....	27
2.3 MATLAB Simulation Results.....	32
Chapter 3 Digital Back-End On-Chip Implementation	39
3.1 Block-level Implementation.....	40
3.1.1 Decimation filter.....	41
3.1.2 Activity detection unit	42
3.1.3 Baseline calculator unit	47
3.1.4 Clock generator and selector unit	49
3.2 Overall Back-end Integration and Implementation.....	50
3.3 Physical Layout.....	54
3.3.1 Power overhead of Digital Back-end.....	55
3.4 Chip Fabrication and Measurement Setup	56

Chapter 4	Mixed-Signal Front-End Implementation.....	58
4.1	Digital-to-analog converter (DAC).....	60
4.2	Integrator circuit (Gm -C stage).....	67
4.3	Voltage comparator.....	74
4.4	Simulation results.....	75
Chapter 5	Conclusions and Future Directions.....	81
5.1	Conclusion.....	81
5.2	Statement on contributions.....	83
5.3	Future work.....	83
5.3.1	Improvement of the non-ideal effects on the front-end.....	84
5.3.2	Improvement on the signal transient recovery.....	84
5.3.3	Low power transmitter and data packaging.....	85
5.3.4	Multi-channel implementation.....	85
Appendix A	86
A.1.	MATLAB code of the described system.....	86
A.2.	Verilog code for FPGA board to generate inputs for chip measurement.....	96
A.3.	Description of the measurement setup.....	101
Bibliography	103

LIST OF TABLES

Table 1.1 Power breakdown of implantable BMIs illustrating the Tx's significant share of the total power consumption.....	5
Table 1.2 Tx power consumption of implantable BMIs in various works	6
Table 2.1 Δw for the four possible outcomes based on $v[n]$ and its previous value $v[n-1]$	31
Table 2.2 Statistical summary of the simulation results presented in Figure 2.11.	36
Table 2.3 Statistical summary of the simulation results presented in Figure 2.12.	37
Table 3.1 Digital back-end input values	51
Table 3.2 Digital back-end's specifications.....	54
Table 4.1 Specifications of the neural front-end.....	60
Table 4.2 Size and bias points of the DAC's OTA.....	66
Table 4.3 Size and bias points of Gm stage amplifier.....	70
Table 4.4 Size and bias points of the CF amplifier.....	71
Table 4.5 performance parameter of the Gm – C stage	73
Table 4.6 Performance comparison with the state of the art.....	80

LIST OF FIGURES

Figure 1.1 Comparison between temporal resolution vs. spatial resolution and coverage of several brain monitoring techniques [10]	2
Figure 1.2 Top-level block diagram of a typical wireless BMI and its envisioned interfacing with the brain cortex.	4
Figure 1.3 Simplified block diagram of different generic and application-specific	7
Figure 1.4 EEG signal sparsity in wavelet domain [15]	8
Figure 1.5 Block diagram of a typical CS transmitter and receiver	9
Figure 1.6 EEG signal with events periods marked as A and idle periods as B	11
Figure 1.7 Concept of adaptive sampling	12
Figure 1.8 An example of missing short high-activity periods in adaptive sampling	12
Figure 1.9 Conceptual difference between adaptive sampling and adaptive resolution techniques	13
Figure 2.1 An example of (a) high-activity event (b) noise interferences that unwantedly are detected as a high-activity event[39]	16
Figure 2.2 (a) conventional and (b) direct-ADC recording front-end concepts.	19
Figure 2.3 Comparison between adaptive sampling and adaptive resolution techniques.....	20
Figure 2.4 Internal block diagram of the digital backend blocks of the proposed adaptive-resolution neural ADC.	24

Figure 2.5 Output of the N-bit U/D counter; counter is designed for the high resolution; the lower resolution is achieved by increasing the UP/DOWN counting step size.	26
Figure 2.6 system-level diagram of the proposed Δ - $\Delta\Sigma$ front-end architecture	29
Figure 2.7 Block diagram of the proposed Δ - $\Delta\Sigma$ ADC	29
Figure 2.8 w results for random output bit sequence (for w calculation bit 0 is equivalent to -1)	31
Figure 2.9 Frequency response of the proposed mixed signal front-end in MATLAB, for 4-bit and 8-bit resolution from left to right	34
Figure 2.10 EEG signal (top), the reconstructed output with adaptive resolutions (middle), and the automatically calculated baseline (bottom).....	35
Figure 2.11 An example of system’s response to an EEG signal with seizure.....	36
Figure 2.12 An example of the proposed system’s response to an EEG signal containing a seizure episode.	37
Figure 3.1 Top-level block diagram of the proposed recording channel and the detailed implementation of the digital blocks.....	40
Figure 3.2 Logic of decimation filter.....	42
Figure 3.3 Threshold levels and “CLOCK SLECTOR” flag value versus signal magnitude variations.....	43
Figure 3.4 Simplified RTL implementation of the activity detection unit: (a) Setting mid-range threshold levels and signal position with respect to the high/low thresholds (b) determining the new value of flag “FLAG_LOW_BAND” and “FLAG_HIGH_BAND” (c) determining “CLOCK SELECTOR” value.....	45

Figure 3.5 Flags behavior for the activity detection unit with respect to threshold levels, figure only depicts for high threshold levels, same behavior for low threshold levels	46
Figure 3.6 Internal block diagram of the baseline calculator unit	47
Figure 3.7 Logic of clock generator & selector unit.....	49
Figure 3.8 Overall block diagram of system.....	50
Figure 3.9 Simulation results for clock generator unit	52
Figure 3.10 Simulation results for activity detection and decimation filter unit	52
Figure 3.11 Simulation results for baseline calculator unit	52
Figure 3.12 Chip's layout and the micrograph	56
Figure 3.13 Lab set up for chip's measurement test	57
Figure 4.1 Top level block diagram of the presented recording channel.....	59
Figure 4.2 (a) Discrete time (DT) and (b) continuous-time (CT) 2nd order $\Delta\Sigma$ modulator block diagram with input arrangements that result in second-order noise shaping and first-order signal shaping (i.e., Δ modulation). Proposed mixed-signal front-end architecture	61
Figure 4.3 Internal architecture of the variable-step integrating-summing DAC	63
Figure 4.4 Telescopic OTA used in the DAC circuit.....	65
Figure 4.5 Frequency response of the designed OTA.....	66
Figure 4.6 Detailed schematic of the Gm-C circuit	68
Figure 4.7 Detailed schematic of the (CMFB) circuit that is shown as; A_CF in the Gm-C circuit.	71
Figure 4.8 Frequency response of the designed G_m-C stage.....	72
Figure 4.9 Power spectral density of the IRN of the designed Gm – C stage.	73

Figure 4.10 Transistor-level schematic of the StrongArm voltage comparator used in the proposed mixed-signal front-end.	74
Figure 4.11 Block diagram of the $\Delta\Sigma$ modulator.....	75
Figure 4.12 VREFN delayed version of the input (VINP) made from the Modulator’s output using the proposed DAC.....	76
Figure 4.13 Step size integration according to modulators output bit stream sequence.....	77
Figure 4.14 Input DC drift cancelation through the proposed DAC architecture.....	78
Figure 4.15 FFT of the modulator’s output for OSR=16 (4-bit resolution)	79
Figure 4.16 FFT of the modulator’s output for OSR=64 (8-bit resolution)	79

LIST OF ACRONYMS

Acronym	Meaning
MEG	Magnetoencephalography
PET	Positron Emission Tomography
MRI	Magnetic Resonance Imaging
iEEG	Intracranial EEG
EEG	Electroencephalography
ECoG	Electrocorticography
ECG	Electrocardiogram
EMG	Electromyography
DAC	Digital-to-Analog Converter
ADC	Analog-to-Digital Converter
CMRR	Common Mode Rejection Ratio
IRN	Input Referred Noise
SAR ADC	Successive-Approximation-Register Analog-to-Digital Converter
CT	Continuous time
DT	Discrete time

Chapter 1

Introduction and Motivation

1.1 Motivation and objectives

Neurological disorders affect more than one billion people worldwide today, and the number is expected to increase with the world's aging population [1]. Accurate detection and effective control of these disorders require continuous monitoring of brain neuronal activities with high spatial and temporal resolution. Over the past decade, thanks to the advancements in the fields of neurotechnology and microelectronics, various implantable and wearable brain machine interfaces (BMI) have been developed to monitor, diagnose, and control different neurological disorders [2-9,47-50].

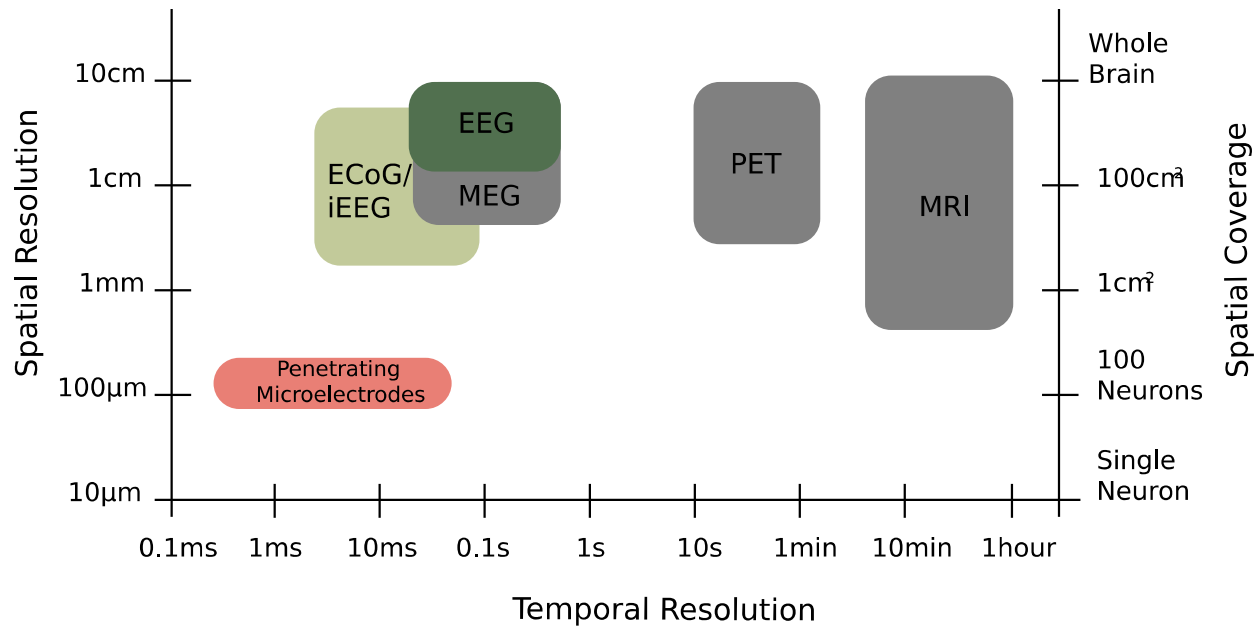


Figure 1.1 Comparison between temporal resolution vs. spatial resolution and coverage of several brain monitoring techniques [10]

In Figure 1.1, various methods for brain activity monitoring is shown. Each of them come at its own advantage and disadvantages. Methods such as magnetoencephalography (MEG), positron emission tomography (PET), magnetic resonance imaging (MRI), while are noninvasive, are not suitable for ambulatory applications due to either of size constraints, portability, power requirements, and poor temporal resolution [10]. Highly-invasive methods such as intracranial EEG (iEEG) using penetrating microelectrodes offer not only high spatial resolution, which is necessary for capturing time-sensitive neurological events (e.g., epilepsy seizures [51]), but also high temporal resolution (i.e., update rate). However, this type of monitoring can cause substantial damage to the brain tissues, thus is only used in severe conditions when all other options are exhausted. On the other hand, less-invasive electrophysiological recording methods such as scalp (or surface) electroencephalography (EEG) [52][53], and electro-corticography (ECoG) allow for monitoring with high temporal resolution. Depending on the application and the required spatial

resolution and spatial coverage, one of these technologies are used for long-term monitoring of the patient's neural activity [10].

1.1.1 Implantable BMIs

Timely and accurate detection of neurological events often requires long term (i.e., months to years) monitoring of brain activity from a widespread network of neurons on the brain. For these situations, a feasible solution is to develop wireless cm-scale BMI devices implanted under the scalp (to be minimally obtrusive for the patient) and connected to an ECoG electrode array with as many recording sites as needed to yield the desired spatial resolution and coverage. Regardless of using a battery or a wireless powering link, the energy budget of these devices is highly constrained either due to physical size limitations for the battery, or the safety limitations for power consumption and power transfer density [54]. For power transfer, the upper limit on the magnetic field intensity is set by the regulations and guidelines on the specific absorption rate (SAR) [11]. Based on these guidelines, the total heat density created by the magnetic field anywhere in the body cannot exceed the rate of 1.6W/Kg [11]. Additionally, heat dissipation within an implantable BMI resulting in one-degree Celsius temperature increase is deemed unsafe for the cortex tissue. This corresponds to an overall system power density of 15-80mW/cm² [12], depending on the heat conductivity of the encapsulation materials.

Figure 1.2 depicts the top-level block diagram of a typical fully wireless implantable BMI and how it could be interfaced with the brain. As shown, a neural recording system consists of an array of recording channels (typically, 64 or more) as well as blocks for signal processing, wireless data transmission and power telemetry [55]. Each recording channel consists of a mixed-signal front end, which is responsible for low-noise amplification, and digitization of the sensed neural data.

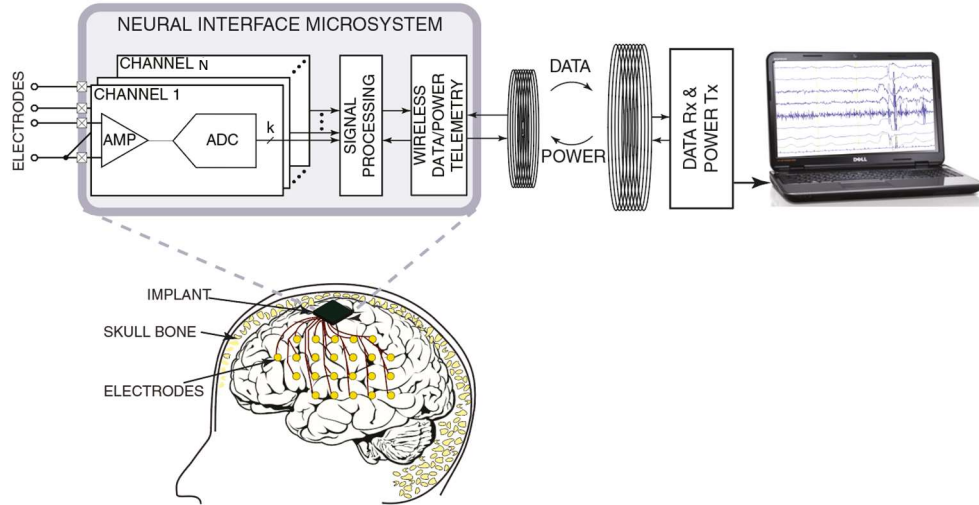


Figure 1.2 Top-level block diagram of a typical wireless BMI and its envisioned interfacing with the brain cortex.

1.1.2 Power constraints for channel count scaling

As mentioned, to improve spatial resolution and coverage, and consequently the accuracy of neurological events detection, it is desired to have as many recording channels as possible. However, besides directly increasing the required power for recording, channel count scaling also increases the throughput required for the wireless data transmitter (Tx) to communicate the recorded data outside of the body. From power budgeting perspective, the increase in the required transmission data-rate is particularly undesirable because wireless data transmission is typically responsible for 70-to-90% of the total power consumption of implantable BMIs [13][7][3][8]. Table 1.1 lists some of the implantable neural interface integrated circuits (ICs) and the share of the total power consumption allocated to data Tx.

Table 1.1 Power breakdown of implantable BMIs illustrating the Tx's significant share of the total power consumption

Ref.	JSSC'17 [13]	JSSC'14 [7]	JSSC'15 [8]	JSSC'16 [3]	TBCAS'16 [14]
Tech.	130nm	180nm	180nm	130nm	180nm
Power/Channel (μW)	0.63	57.67	1.62	9.1	3.2
No. of Recording Channels	64	8	16	64	16
Total Power Consumption (Including TX) (mW)	1.07	2.8	0.25	2.17	0.25
% of TX Power Consumption	~96%	~83%	~89%	~73%	~79%

This motivates for investigating system and circuit techniques to improve energy efficiency of data Tx. Various architectures and time- and frequency-domain modulation schemes have been investigated over the past decade, and energy efficiencies in the range of a few to pJ/bit have been reported, as listed in Table 1.2. It should be noted that less efficient works (i.e., higher pJ/bit) are often yield a better transmission range and fidelity for the same bit error rate. Regardless, the data Tx remains the major power consumer in implantable BMIs and a significant bottleneck for channel count scaling [56]. As such, in parallel with efforts focusing on improving energy efficiency, it is critical to investigate approaches/methods for reducing the amount of data that needs to be transmitted (i.e., the required Tx throughput) without losing recording quality or detection accuracy [57].

Table 1.2 Tx power consumption of implantable BMIs in various works

Ref.	TCAS-II'21 [16]	JSSC'15 [17]	JSSC'14 [18]	JSSC'14 [19]	VLSI'17 [20]
Process	180nm	180nm	180nm	90nm	180nm
TX Modulation	OOK/FSK	OOK	OOK/FSK	OOK	OOK
Data Rate (Mbps)	10	5	5	5	10
Energy/Bit (pJ/bit)	7	19.6	38	172	171

1.2 Data Compression

1.2.1 Introduction and trade-offs

As will be discussed in the remaining of this chapter, there are several ways to reduce/compress the size of data. However, while the ultimate goal is to achieve the highest compression ratio (CR), this must not come at the cost of losing valuable data.

For some applications it is possible to significantly reduce the required Tx's throughput by only transmitting the outcome of on-chip signal processing instead of the raw recorded data. In the extreme case, the entire signal processing could be done on-chip so that only the result (e.g., a one bit signal showing a classification output) needs to be transmitted. While this has the potential to reduce the data size by orders of magnitude and practically solves the problem, as discussed in [40], for many cases it is desired to transmit the raw (i.e., unprocessed) data (e.g., when processing algorithm is too computationally expensive to be done on-chip, or for physician's future review), demanding the data compression technique not to be specific to an application.

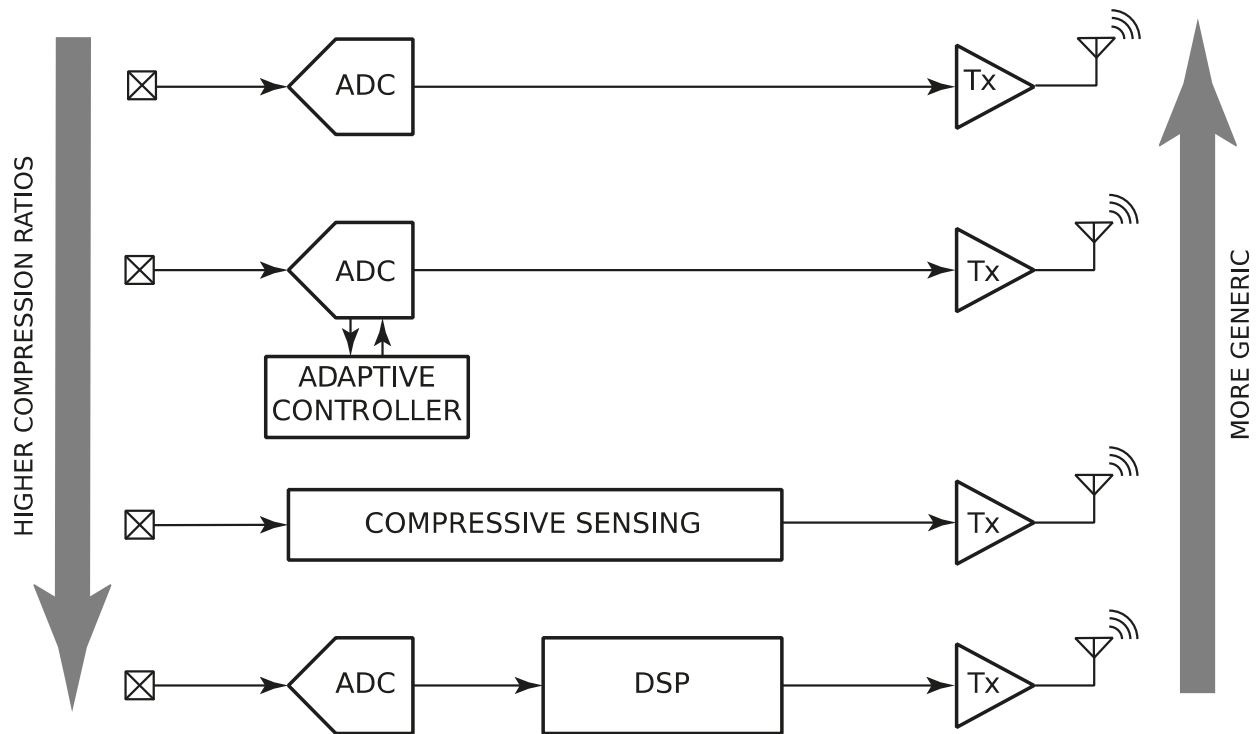


Figure 1.3 Simplified block diagram of different generic and application-specific methods used for data reduction in implantable BMI devices.

Figure 1.3 shows some of the most popular approaches used for data acquisition in neural recording devices, and where each of them stands on the spectrum of achieved compression ratio vs. generality of the approach. Basically, one end of the spectrum is to send the raw data, without performing any form of compression, while the other end is to perform extensive data processing on the acquired data and transmit only specific features of the data, which makes it the least generic. Techniques such as compressive sensing or adaptive compression (e.g. adaptive sampling) stand somewhere in the middle of this spectrum. As will be discussed, among the two, compressive sensing achieves a higher CR but is less generic compared to adaptive compression techniques. Besides CR, data loss, and generality of the technique, reconstruction accuracy in the receiver, and the required on-chip computational power for conducting the compression are critical parameters that must be considered when choosing data compression method.

1.2.2 Information sparsity

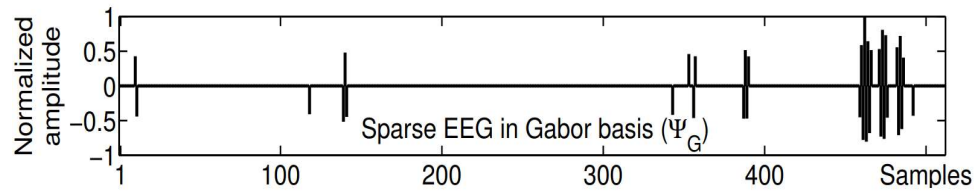


Figure 1.4 EEG signal sparsity in wavelet domain [15]

In many applications, data sparsity is the key characteristic that is widely leveraged of in many compression techniques. A data is said to be sparse in a certain domain (i.e. time, frequency) when most of the signal's magnitudes in that domain, are either zero, or insignificant. As an example, Figure 1.4 shows an EEG data is sparse in Gabor basis, as it contains very few non-zero values; or a sine wave is said to be sparse in frequency domain, as it only has one non-zero component in that domain.

Sparsity in bio-signals can be defined in an another way as well. A bio-signal is information sparse meaning, in time-domain, biologically meaningful events (e.g., epileptic seizures in an iEEG recording) are inherently unpredictable and happen infrequently.

1.2.3 Compressive sensing

Compressive sensing (CS) is a compression technique used widely in applications such as image processing, radar, video coding and many more [23] [24] [25]. This method can be considered more on the generic side of the spectrum, since it can be performed on any signal that have sparsity in at least one domain. A key advantage of this technique is that it allows for power-friendly implementation as in this method, compression is done in the time domain, regardless of the domain of sparsity.

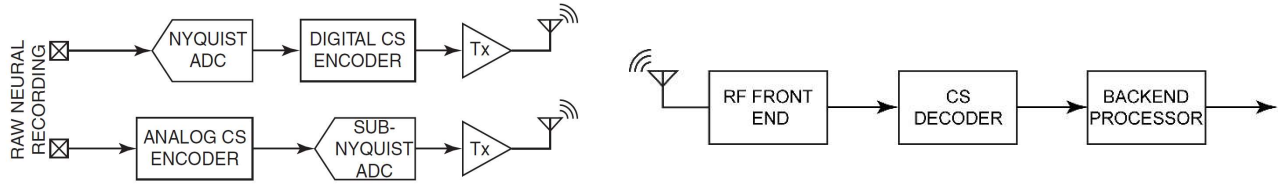


Figure 1.5 Block diagram of a typical CS transmitter and receiver

In this method, unlike conventional data acquisition methods, in which data is acquired and transmitted at a rate at least two times the highest frequency component of the signal (the Nyquist-Shannon rate), data is compressed and reduced proportional to the amount of information it carries in its domain of sparsity [26], promising significantly lower amount of data needs to be transmitted for lossless signal reconstruction [26] [27].

CS is conducted by projecting the Nyquist rate sampled data X , which holds $X_{N \times 1}$ samples, to its compressed version $Y_{M \times 1}$, which will have M samples, where $M \ll N$. The projection is done through multiplication of $X_{N \times 1}$ by a “measurement matrix” $\varphi_{M \times N}$, resulting in a compressed version of $X_{N \times 1}$ called $Y_{M \times 1}$.

Implementation of the encoding circuit can be done in either digital or analog domain (Figure 1.5), using a set of mixers and integrators, which despite seemingly complicated computational procedure of the method, makes the technique power friendly for on-chip implementation. Works done in [29], [30] have implemented the technique in the digital domain, whereas, works done in [26], [28], [31] offer an analog based compressed sensing. Generally, for wireless sensor application, it is shown that digital based compressive sensing implementation achieves better power efficiency [27].

In this technique, data reconstruction procedure consists of a set of complex power hungry computations. However, for implantable BMIs, receiver is outside of the body and has far less

constraints in terms of computational power and energy budget [26], so this might be a less critical disadvantage.

The key drawback of CS is that conducting any non-trivial signal processing on compressively-sensed signal is infeasible before it is decompressed (i.e., decoded) in the receiver. In many implantable devices developed for diagnostic and closed-loop responsive treatment, it is critical for the BMI to conduct signal processing on chip (e.g., to minimize the processing delay) [2][13][32][38]. However, the compressed signal is far too modified to perform any signal processing on it.

1.2.4 Adaptive sampling

Adaptive sampling is another popular method of compression which relies on the sparsity of events occurrence in bio signals. As shown Figure 1.6, during periods "A", a significant event happens, whereas during periods "B", signal is almost idle and holds no neurologically-relevant information. Considering the high likelihood of sparsity in bio-signals, adaptive sampling suggests adjusting the sampling rate in real time and according to the signal's level of activity. This allows for sampling with Nyquist rate during periods with high level of activity and with a sub-Nyquist rate for the rest of time. Considering the overwhelming dominance of the "idle" periods compared to the high-activity periods, this results in a significant reduction in the acquired data volume, and consequently, the required throughput for wireless Tx. The word adaptiveness comes from the fact that, each BMI's recording channel continuously adjusts its sampling rate according to signal's level of activity, therefore, making the power consumption, also adaptive to the events occurrence.

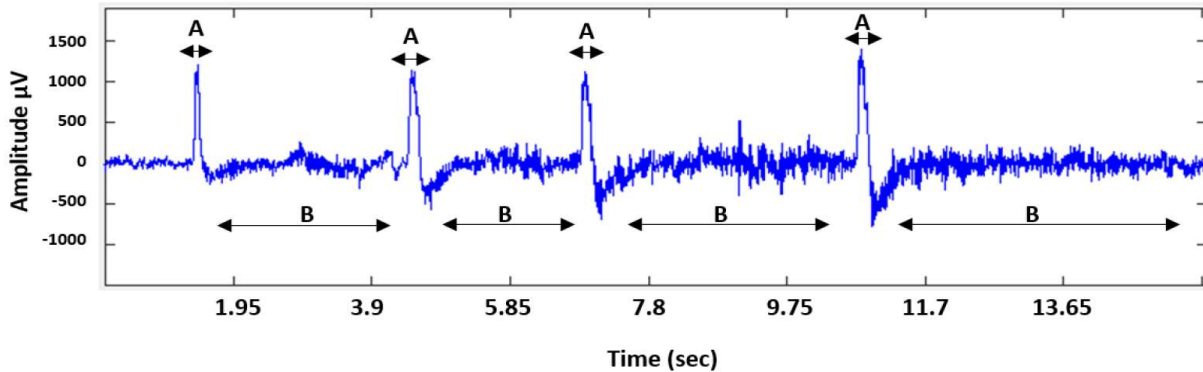


Figure 1.6 EEG signal with events periods marked as A and idle periods as B

An important challenge in adaptive sampling is accurate identification of high-activity periods. Thresholding is one of the methods in which the incoming signal's amplitude is compared to a set of pre-set thresholds to evaluate the magnitude of deviation from the signal's baseline and adjust the sampling rate accordingly, as illustrated in Figure 1.7. This method's performance is highly sensitive to proper selection of threshold values. A too-small threshold value could lead to identifying the majority of the recording as "high activity", hence achieving little to no data compression, while a too-large threshold could result in missing important neurological events.

The relatively-high chance of missing all or part of a neurological event is in fact one of the major drawbacks of adaptive sampling. This is due to the fact that sampling rate is set according to the N previous samples. Therefore, for any event that follows a long period of idleness, there is a high chance that part or all of it is missed due to the latency in adjusting sampling rate. Figure 1.8 illustrates this drawback graphically. As shown, while events with long period are almost fully captured, the shorter events are completely missed, making the method unreliable for applications where missing even one event could lead to severe consequences.

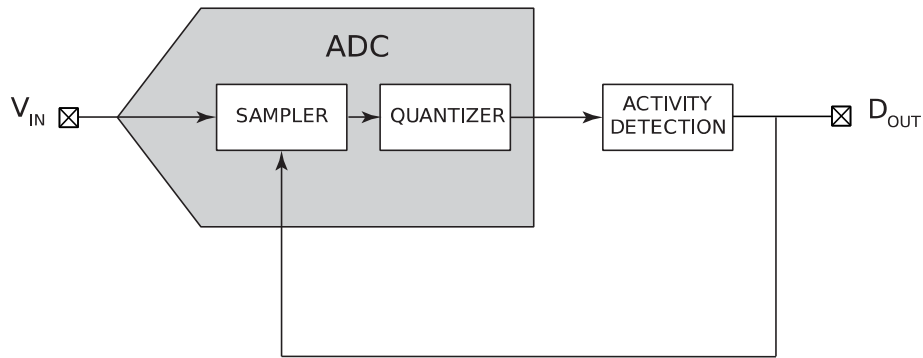


Figure 1.7 Concept of adaptive sampling

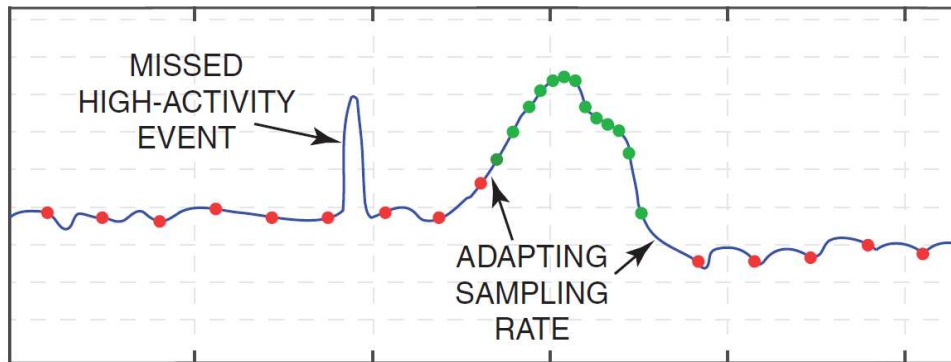


Figure 1.8 An example of missing short high-activity periods in adaptive sampling

1.3 Adaptive resolution

The mentioned problem with adaptive sampling motivates us to look for another compression technique which can eliminate the risk of information loss, while still being able to leverage from signal's information sparsity, i.e., adaptively adjust the data rate according to the amount of vital information signal holds. The technique that is proposed here, is adaptive resolution compression. In this technique unlike adaptive sampling, sampling rate is always kept at Nyquist rate, and alternatively only the resolution of the sampled data varies. This way, while the bit rate is reduced, there will be no risk of losing either short or long-period events. Figure 1.9 depicts the difference

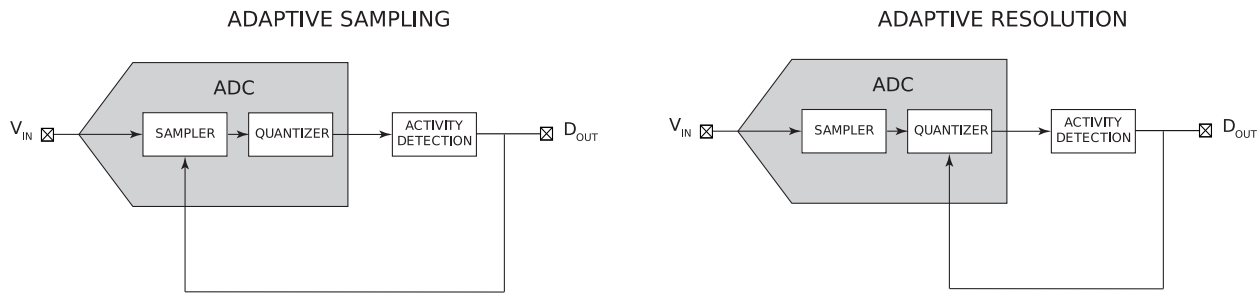


Figure 1.9 Conceptual difference between adaptive sampling and adaptive resolution techniques

between the concept of adaptive sampling, which was explained in section 1.2.4, and adaptive resolution discussed here. As shown, the main difference here is that the feedback signal which is the result of event detection, is fed to ADC's quantizer rather than the sampler, resulting in changing the quantization resolution and keeping the sampling rate at Nyquist. While the difference between the two methods looks minor and simple, implementing an adaptive resolution scheme introduces several challenges and complications in terms of designing a compatible front-end, efficient design of a variable data converter, a loss-less event detection unit, and a low-latency resolution adaptation controller. It should be mentioned although the focus of this work is on neural signals, however, this compression technique is applicable for any sort of signal containing some sort of information sparsity or event sparsity in time-domain. As an example, many types of events in bio-signals other than neural signals such as; ECG, EMG, etc., show this type of sparsity and therefore the technique generality is not only applicable to various type of neural events but also it is expandable to other types of bio-signals.

It should be mentioned that in addition to power saving through reducing total number of transmitted data, the constraint on bandwidth through this compression will be relaxed as well. To elaborate, because of reduction in overall number of data for a fixed duration compare to when the

technique is not used, data can be transmitted at a lower bit rate and therefore reducing the required bandwidth for transmission.

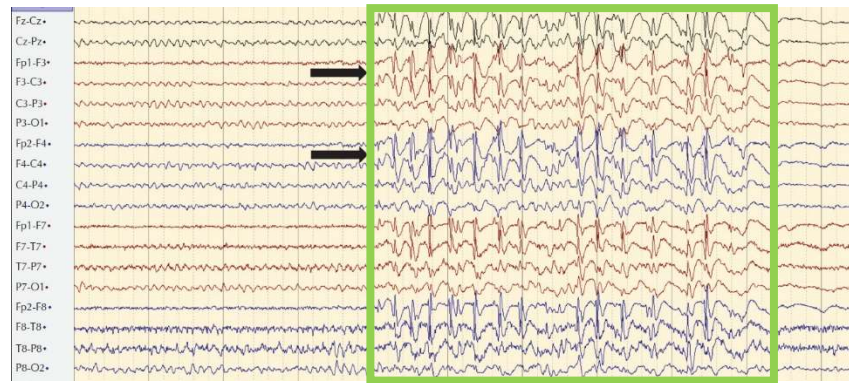
1.4 Thesis Organization

Chapter 2 will focus on the system-level implementation of the adaptive resolution compression technique, and includes system-level simulation results illustrating the proposed idea's efficacy in improving the recording system's energy efficiency. Chapter 3 will discuss the detailed implementation of the digital backend modules responsible for high-activity event detection, signal baseline calculation, and responsive resolution adjustment. These modules are integrated on a silicon IC and fabricated. The physical layout and experimental setup for this IC are also presented in this chapter. Chapter 4 will discuss the design procedure, detailed transistor-level implementation, and system- and circuit-level verification of a novel mixed-signal fully-dynamic-power neural ADC developed as the recording front-end module for the presented system.

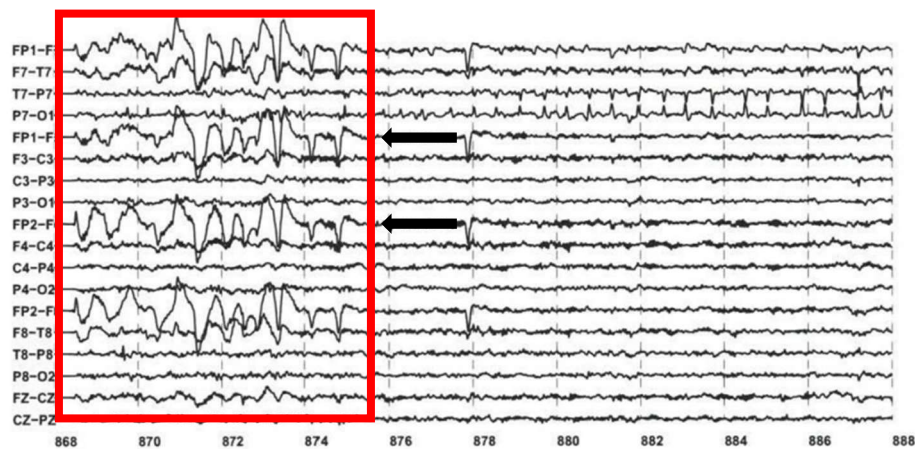
Chapter 2

System-Level Implementation

As discussed in chapter 1, the main focus of this work is to improve the energy efficiency of the implantable device by adapting its recording accuracy to the input signal's level of activity, without any meaningful loss of data. In this chapter, in order to understand the principles of the proposed idea, first, a few key definitions will be explained, and then the working principles of the mixed-signal front-end and the digital backend of the proposed architecture will be presented. In the end, the MATLAB simulation results and the potential efficacy of the proposed technique will be discussed.



(a)



(b)

Figure 2.1 An example of (a) high-activity event (b) noise interferences that unwantedly are detected as a high-activity event[39]

2.1 Background Theory

2.1.1 Events in Neural Recording

In this work, an event (or interchangeably a “high-activity event”) is defined as when the time-domain amplitude of the neural signal experiences a considerable (i.e., above a predefined threshold) deviation from the "baseline", which is the average signal level during idle (i.e., no event) periods, Figure 2.1 (a). While these "high-activity" episodes could be due to many reasons such as background noise, interference, motion artifacts, stimulation artifacts, etc., their

appearance could also be associate with neurologically-relevant events such as an epileptic seizure, as shown in Figure 2.1 (b). If the threshold for identifying a period of the signal as a "high-activity event" is set low enough, it could be claimed with high certainty that the idle periods do not contain any critical information about neurologically-significant episodes, hence, capturing them with low accuracy will not result in any loss of critical information. Considering the information sparsity of neural signals in the time domain (in terms of the occurrence rate of "high-activity events"), this allows for a significant reduction in the required power for data acquisition as well as the required throughput for data transmission. Of course, the success in both keeping the data integrity and power/resource reduction heavily depends on how well the event thresholds are set. We will discuss the strategy and mechanism for threshold selection in due course.

2.1.2 Neural ADCs

The main tasks of the circuits designed for recording brain's neural activity are low-noise signal amplification and quantization. Once a neural signal is amplified and digitized, various types of signal processing could be performed on it. This is why conventional neural recording channels consist of a low noise amplifier and an ADC, as shown in Figure 2.2 (a). Despite the early success of this conventional approach, it was shown not to be suitable for massive integration (i.e., hundreds or thousands of recording channels on one chip) mainly due to the required silicon area for its implementation [13].

To overcome the power scalability issues, various types of ADC are proposed for neural recording such as nonlinear, predictive or level-crossing ADCs. In [44] a nonlinear ADC introduced in which through an interior anti-log ADC, non-uniform quantization is achieved to reduce the amount of bit rate transmission. In [45], a predictive scheme is used to implement the

predictive ADC. In this ADC by predicting the incoming signal through the previous samples, incoming sample digital conversion in the SAR ADC is switched and done for a sub-range instead of digital full-scale range. This way, unlike previous ADC which reduces the power consumption through the reduction of data rate, the ADC power itself is scaled by burning lower power for when conversion is done in sub-range. In [46] another popular type of ADC called level-crossing ADC is used, in which output is generated each time signal passes a pre-set level along with the timing duration between two consecutive samples. The sampling in this type of ADC is done in a non-uniform way and its power consumption depends on the rate of signal's variations.

Another type of ADC which has become popular in recent years for neural recording channels is neural ADCs or direct ADC architectures. In this type of neural recording front-ends (shown in Figure 2.2(b)), unlike conventional ones, both processes of signal amplification and digitization are performed in a single oversampling ADC stage. It has been shown that by avoiding the analog low-noise amplifiers and relying on the mixed-signal architecture of these ADCs, high-precision neural recording could be performed without the need for bulky non-scalable passive components (e.g., input capacitors used for AC coupling) [32]. Additionally, these oversampling ADCs are capable of achieving very high resolutions (e.g., >14 bits), at a reasonable power budget, particularly for applications that have a low input frequency bandwidth (i.e, a few kHz) such as neural signals. This allows for achieving a very high dynamic range for the recording circuit, which is a critical requirement if simultaneous recording and stimulation is to be conducted [33][58]. Most importantly for the purpose of this work, the power consumption of direct ADC architectures is fully dynamic, unlike the case for conventional architectures where the majority of the recording channel's power consumption was due to the static power of the low noise amplifier. This means

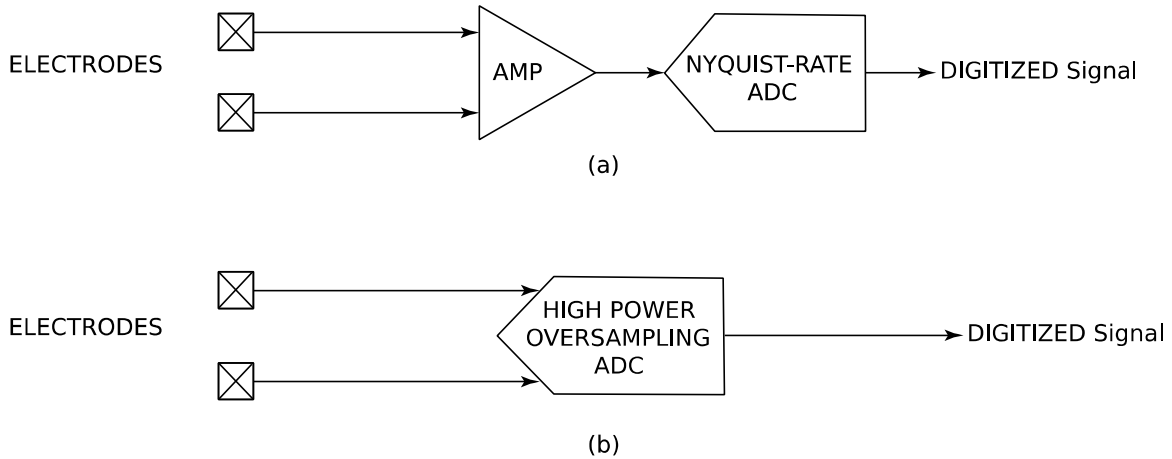


Figure 2.2 (a) conventional and (b) direct-ADC recording front-end concepts.

that if the ADC's precision, hence its power consumption, is made adaptive to the input signal's level of activity, the power saving benefit is applicable to the entire recording channel, and not just a small portion of it (as in conventional architectures).

2.1.3 Adaptive Resolution vs. Adaptive Sampling

Prior to discussing implementation, it is important to clearly distinguish between the widely known concept of adaptive sampling and the proposed concept of adaptive resolution.

As shown in Figure 2.3(a), in adaptive sampling, the controller adjusts the system's, sampling rate to capture the input signal with at least two different sampling frequencies of $f_{S_{low}}$ and $f_{S_{high}}$, with $f_{S_{high}}$ being a super-Nyquist frequency. If the controller recognizes that the input signal holds vital information, system will switch from $f_{S_{low}}$ to $f_{S_{high}}$, and since $f_{S_{high}}$ is set to at least the Nyquist-rate ($f_{S_{high}} \geq f_{Nyquist} = 2f_{IN_{MAX}}$), all the information within the signal is captured. If the controller doesn't detect an upcoming important event, the system's frequency is switched back to $f_{S_{low}}$, which is set at a sub-Nyquist-rate ($f_{S_{low}} < 2f_{IN_{MAX}}$), and therefore, signal is not

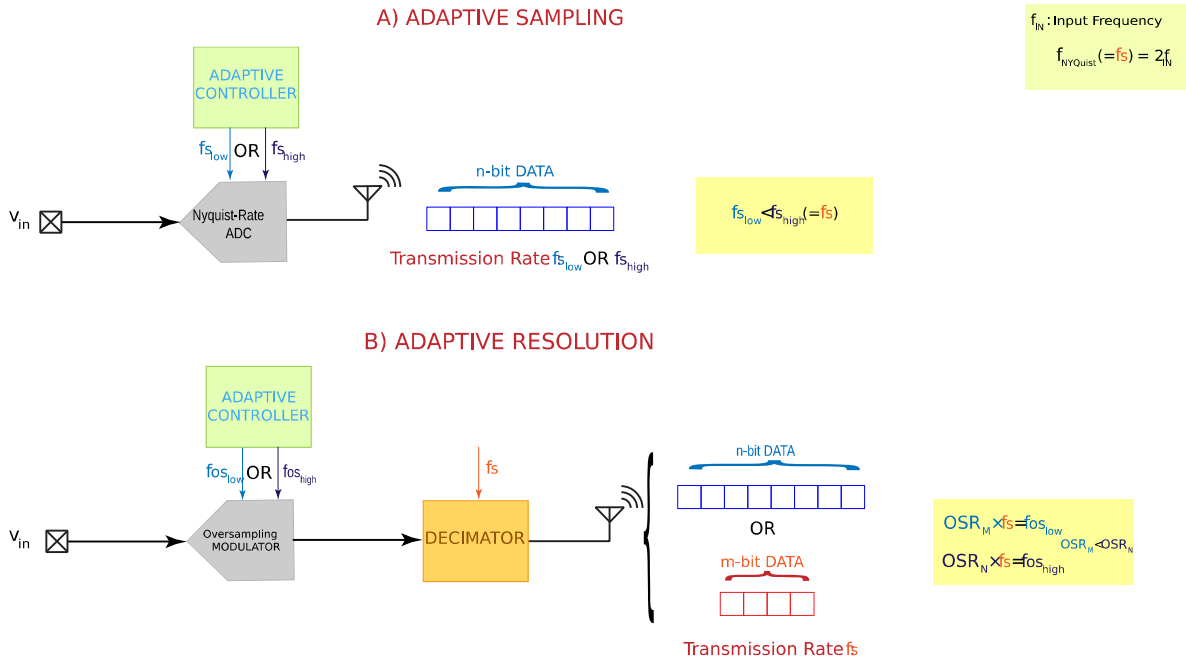


Figure 2.3 Comparison between adaptive sampling and adaptive resolution techniques

fully captured. This data loss is acceptable since the input signal during idle periods is not carrying valuable information.

As shown in Figure 2.3(a), the transmitter also sends data packages according to the two sampling rates. Therefore, for the idle periods, which are expected to be the majority of the neural recordings, the transmission rate, hence the transmitter's power consumption is reduced according to the sampling rate reduction. As discussed in Chapter 1, since transmitters are the most power consuming block of the implantable device, this power saving is significant and could enable the integration of many more recording channels on the chip for the same power budget. However, as it was mentioned in previous chapter, this method is not the best compression technique candidate for situations where there are short-period high-activity events, as system could completely miss that type of activity due to sub-Nyquist sampling, which could have dire consequences for devices with diagnostic applications.

The aforementioned problem, motivated the proposed technique called adaptive resolution. In this technique, instead of varying the sampling rate, quantization resolution adapts to the input signal's level of activity. Figure 2.3(b) depicts a high-level implementation of this idea using an oversampling ADC, which are particularly advantageous for implementation of a variable-rate data converter because they require the same low-resolution quantizer (e.g., in most cases a single-bit quantizer such as a voltage comparator) irrespective of their targeted resolution. Indeed, the quantization resolution in these ADCs are set by the oversampling ratio (OSR) and the order of the loop filter. As such, by changing the clock frequency of the modulator, and without changing the quantizer, the quantization resolution could be varied dynamically. As such, the resolution variability comes at almost no extra power or area cost and no component overdesign.

In contrast, for Nyquist-rate ADCs (e.g., a SAR ADC), the quantizer architecture and its components specifications, particularly those with static power consumption, (e.g., settling time of an OpAmp used for successive approximation) are set by the highest targeted resolution and reducing the resolution will not result in a proportionate power reduction. This comes at the cost of extensive area and power overdesign, since for most of the time the recording system operates in low resolution mode, due to the sparsity of high-activity events in neural signals.

In addition to the aforementioned problem, in Nyquist-rate ADCs, the resynchronization and reconstruction of the sampled data with different resolutions is a complex task in the receiver, whereas in oversampling ADCs, all that needs to be done is to sync the decimator with the variable OSR, as will be described later.

In Figure 2.3(b), the incoming data (V_{IN}) is either sampled with a high oversampling frequency ($f_{OS_{high}}$) or a low oversampling frequency ($f_{OS_{low}}$). In an oversampling ADC, there is a direct relationship between OSR and oversampling frequency ($f_{Nyquist} = 2f_{IN_{max}}$, $f_{OS} = OSR \times$

$f_{Nyquist}$). Also, there is a logarithmic relationship between the oversampling frequency and SQNR as shown in (2.1)

$$SQNR = 3.01 \times K \times (2L + 1) - 9.36L - 2.76, \quad (2.1)$$

Where

$$OSR = 2^K \text{ and } L = \text{Order of modulator}$$

Therefore, by modifying the OSR, different sampling resolutions can be achieved. As shown in the figure, prior to being fed to the wireless transmitter, the output of the modulator is passed through a decimation filter resulting in the removal of high-frequency out-of-band noise and reducing the data rate to the Nyquist-rate. In this way, regardless of the resolution, unlike adaptive sampling, data packages are sent at Nyquist-rate ($f_{nyquist}$), and only size of each package is modified. Here, since data is oversampled with 2 different resolutions, there are some consideration that should be taken into account in designing the decimation filter, which will be discussed in the next section.

It should be noted that while we described both adaptive sampling and adaptive resolution concepts with only two levels (for the sake of simplicity), both techniques could be implemented in a more sophisticated way where multiple levels of sampling rate or resolution are employed.

Adopting the proposed strategy can significantly reduce the required transmitter's throughput, hence, the wireless transmitter power consumption. Applying this method to each recording channel, can further reduce the system's overall power consumption, while assuring all neurologically-relevant events are captured.

2.2 Functional Implementation

Implementation of the proposed technique is divided into two parts, the digital back-end, and the mixed-signal front-end. Each will be explained separately in the following sections.

2.2.1 Digital Back-End

Figure 2.4 depicts the top-level block diagram of the proposed adaptive-resolution neural ADC and the detailed block diagram of the digital backend units, which are the decimation filter (i.e., the N-bit Up/Down Counter + Down Sampler) and the adaptive controller. As shown, the controller receives the ADC's output as the input and decides for one of the two clock frequencies (i.e. $f_{OS_{high}}$ and $f_{OS_{low}}$ in Figure 2.3 (b)) to be selected as the output.

The controller itself consists of an activity detection unit for identifying high-activity events in the neural signal, a Clock Selector unit for adjusting the oversampling frequency based on the input signal's level of activity, and a Baseline Calculator unit for tracking the DC level of the input signal and calibrating the activity detection accordingly. In the following section, these units are explained in detail.

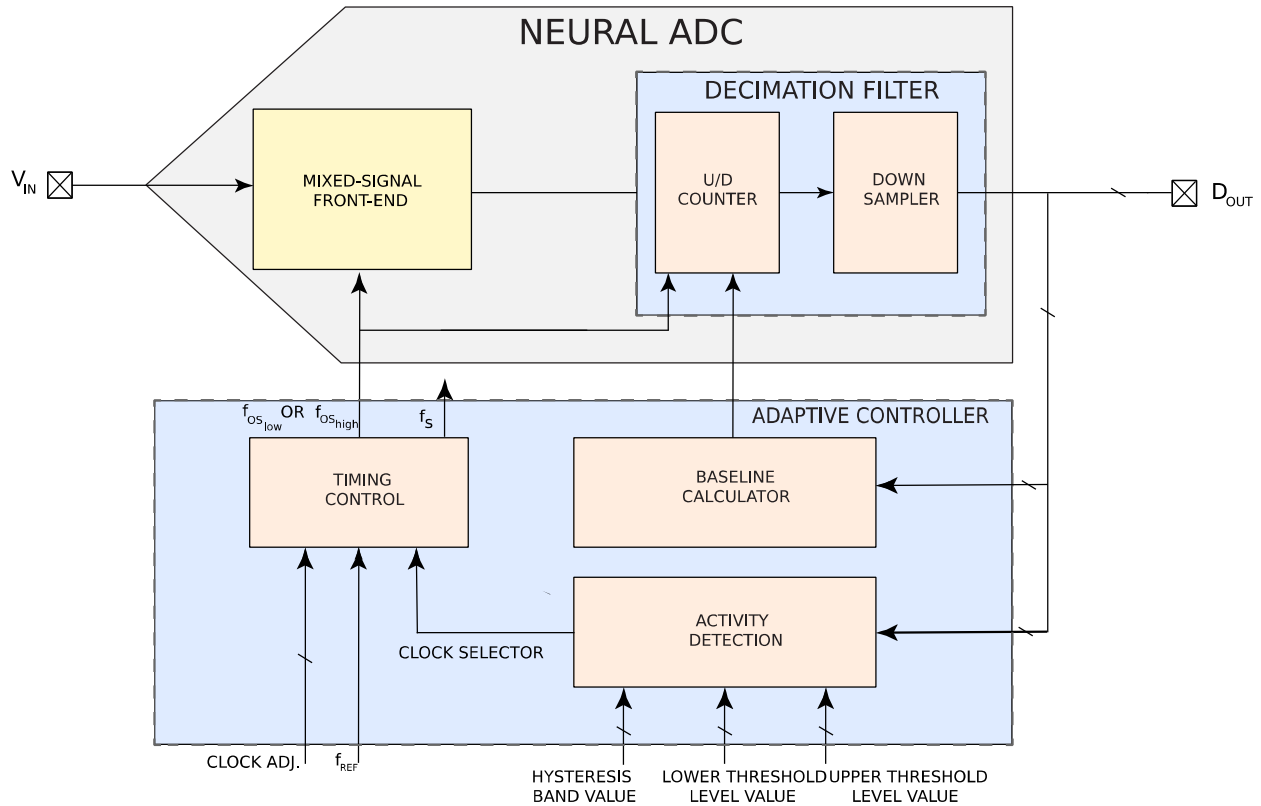


Figure 2.4 Internal block diagram of the digital backend blocks of the proposed adaptive-resolution neural ADC.

2.2.1.a Decimation Filter

As discussed previously, in order to reconstruct signal's quantized amplitude at Nyquist frequency, the modulator's output bitstream needs to get averaged and down sampled (i.e., decimated). In the frequency domain, the combination of these two steps is similar to applying a low-pass filter with a cut-off frequency at the signal's bandwidth ($f_{IN_{MAX}}$). In the presented system, the decimation is implemented by (a) using an N-bit up/down (U/D) counter, which its output value increases/decreases based on the modulator's bitstream (i.e., increases with a 1, and decreases with a 0), and (b) reading out the counter output every OSR clock cycles. This is equivalent to applying a moving average filter to a series of one-bit numbers (modulator's output bitstream which has a bandwidth of $f_{OS} = OSR \times 2f_{IN}$), and having an output for every OSR input value.

Given that for each resolution there is a specific OSR value to be able to decimate different resolutions, the U/D counter should be designed for the highest aimed resolution (refer to figure 2.6(b), in which N is the highest resolution bit count). For lower resolutions, along with oversampling frequency, counter's counting step size also should be adjusted, meaning that if for high resolution, counter increase or decrease its output value by 1 bit, for low resolution case, counter needs to increase/decrease the value by 2^M , where M is the lower resolution bit count (refer to Figure 2.3 (b)).

A down sampler reads the output of the counter at the corresponding OSR rate (OSR_M for low oversampling rate or OSR_N for high oversampling rate), as shown in Figure 2.3(b). In addition, signal's baseline in analog domain (i.e. 0V) is the middle value of counter in digital domain, i.e., for an 8-bit counter, digitized signal's baseline is 128(i.e., 8'b10000000). This way negative voltage values in analog domain will be translated into 0-127 range (i.e., 8'b00000000 to 8'b01111111) of counter's range, and for positive values, from 129 to 255 (i.e., 8'b10000001 to 8'b11111111). Figure 2.5, depicts the explained mechanism more clearly.

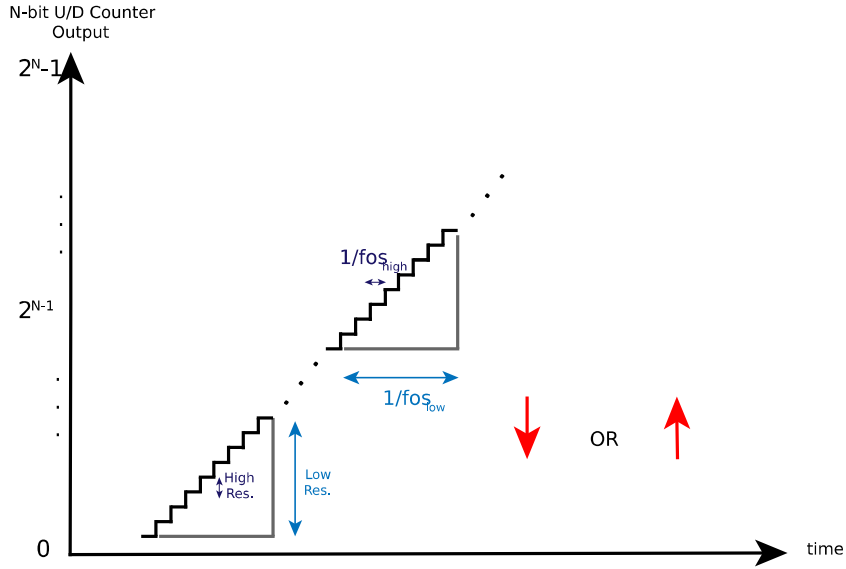


Figure 2.5 Output of the N-bit U/D counter; counter is designed for the high resolution; the lower resolution is achieved by increasing the UP/DOWN counting step size.

2.2.1.b Baseline Calculation

A baseline calculator unit is also designed and included in the system that continuously calculates the DC level of the input signal and adjusts the signal fed to the activity detection unit to have a relatively constant DC level. The baseline is calculated as,

$$DC_{new} = \frac{1 \times C + W \times DC_{old}}{1 + W} \quad (2.2)$$

where C represented the output of the N-bit U/D counter, DC_{old} is the previously calculated DC value, and W is an adjustable coefficient that decides how much weight we assign to the current DC value compared to the new data point (i.e., C) in calculating the new DC value. It should be mentioned that the baseline calculation is done every "A" cycles, where A is also adjustable by the user depending on the variability of the input signal.

The obtained DC_{new} is then subtracted from the counter's output, to maintain a relatively constant DC level for the signal fed to the activity detector. This unit is crucial as the down sampled

signal is compared to a set of fixed threshold levels. By adjusting the signal's baseline before feeding it to the activity detector, we effectively turn the fixed thresholds into dynamic threshold values that adjust themselves according to the signal's DC level.

2.2.1.c Activity Detection

As mentioned in section 2.1, the high-activity events detection is done using dynamic thresholding. Also, as discussed, accurate selection of the threshold values is critical in striking a better trade-off between minimizing data loss (i.e., no event loss) and maximizing the energy efficiency (i.e., maximum data compression). As shown in n Figure 2.4, the decimated data is continuously fed to the activity detection unit and is compared to the pre-set threshold levels by user. In order to prevent unnecessary clock adjustment due to noise/interference that might look like a high-activity event, the system employs a set of hysteresis band, along with the threshold levels, which are customizable by the user.

The output of activity detection unit, which essentially is a one-bit command signal, is send to the timing control unit. The clock selector unit generates the oversampling frequencies ($f_{OS_{low}}$, or $f_{OS_{high}}$) used in the system, and according to the result of the activity, chooses which clock frequency is sent to mixed-signal front-end.

2.2.2 Mixed-Signal Front-End

The mixed-signal recording front-end of the proposed system is a neural ADC, which as mentioned earlier, is built based on principles of oversampling ADCs. By implementing the recording front-end as a neural ADC, besides the inherent benefits of neural ADCs, (e.g., high DR, area and power efficiency, etc.), we leverage the fact that the entire front-end (i.e., amplification and digitization) is implemented in a single block with a fully dynamic power consumption.

Therefore, the proposed adaptive variation of oversampling clock frequency will result in adapting the power consumption of the entire recording front-end, and not just the quantizer.

To put this into perspective, with a conventional recording front-end architecture (i.e., amplifier + ADC), the amplifier has a static power consumption and any adaptivity in the ADC's sampling rate or resolution could only save part of the ADC's power, which is typically 10-20% of the total front-end power. While each recording front-end has a very small power consumption, recent works report close to 100 copies of these circuits integrated in an IC and the research community's target is to integrate hundreds to thousands of recording channel in each IC in the future. Taking that into account, a front-end architecture with fully-dynamic power that is adaptive to the input neural activity seems to become increasingly important in enabling potential scalability of future implantable neural interface microsystems.

The proposed front-end architecture is aimed to achieve a maximum 8-bit recording resolution, while being able to handle large (e.g., 50mV) DC offsets and artifacts at the input. Figure 2.6, depicts the proposed ADC's block diagram, which is a combination of a Δ - and a $\Delta\Sigma$ modulators. Compared to a conventional $\Delta\Sigma$ modulator, an integrator is added in parallel with the feedback path. Therefore, the quantized output is first integrated before being fed back to the subtractor at the input. Due to the inherent low-pass behavior of the integrator in the feedback, its output (\bar{w}) is expected to contain the DC and low-frequency content of the quantized signal. When \bar{w} is subtracted from the input ($u[n]$), the input's DC is blocked and its low-frequency content is attenuated, effectively resulting in a high-pass transfer function for the overall system, which allows for high-frequency content such as the difference between two consecutive input samples

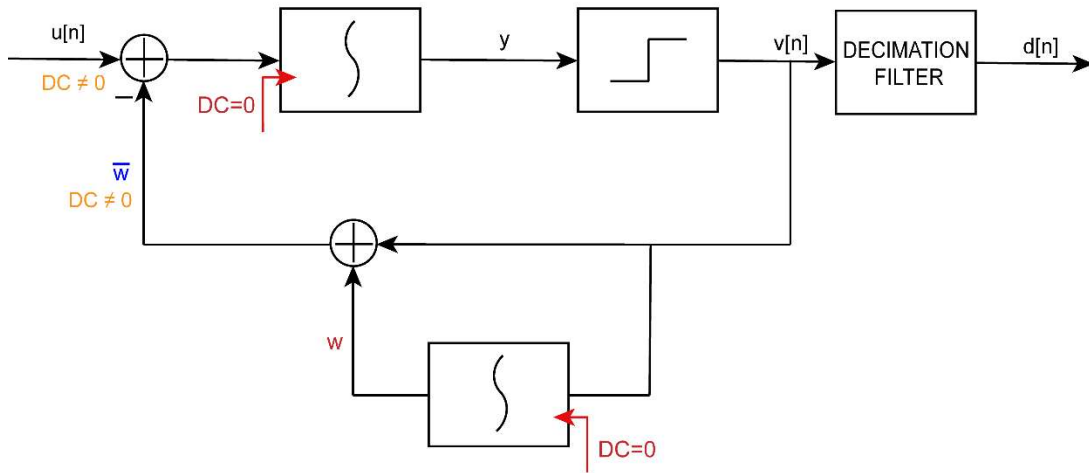


Figure 2.6 system-level diagram of the proposed Δ - $\Delta\Sigma$ front-end architecture

(hence, the additional Δ) to pass. Also, as shown, given that any DC at the input of an integrator results in its output saturation, the negative feedback structure ensures that the DC at the input of both integrators is always maintained at zero.

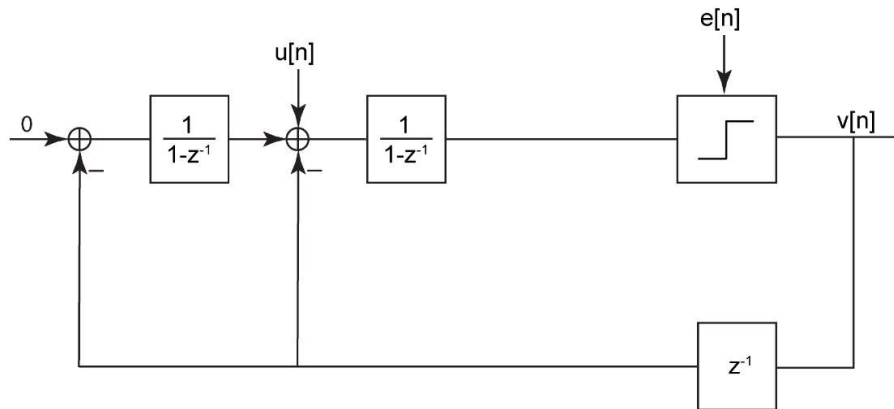


Figure 2.7 Block diagram of the proposed Δ - $\Delta\Sigma$ ADC

More important than the DC removal, the addition of the feedback integrator results in further shaping the in-band quantization noise by another order. Figure 2.7 shows the z-domain block

diagram of the presented ADC. Based on this block diagram, the signal transfer function (STF) and the noise transfer function (NTF) of the presented ADC can be written as,

$$STF(z) = \frac{V(z)}{U(z)} = \frac{\frac{1}{1-z^{-1}}}{1 + \frac{z^{-1}}{(1-z^{-1})^2} + \frac{z^{-1}}{1-z^{-1}}} = (1 - z^{-1}) \quad (2.3)$$

$$NTF(z) = \frac{V(z)}{E(z)} = \frac{(1-z^{-1})^2}{(1-z^{-1})^2 + z^{-1}(1-z^{-1}) + z^{-1}} = (1 - z^{-1})^2 \quad (2.4)$$

The NTF equation shows the second-order filtering of the quantization noise, while the STF shows the Δ modulation and that the signal is differentiated (i.e., $(1 - z^{-1})$), hence its DC is removed.

To efficiently implement the proposed design, the proportional and the integrative feedback paths can be combined into a single path using the algorithm described below. The result of the summation of the two feedback paths in Figure 2.6 for a random output $v[n]$ is shown in Figure 2.8. As annotated in the figure, $v[n]$ represents the quantizer output, w represents the integration of v (increases by one for $v=1$ and decreases by 1 for $v=0$), and \bar{w} represents the sum of the two (i.e., $v+w$).

$$\begin{array}{r}
v[n] = 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots \\
+ \quad w = 1 \ 2 \ 3 \ 2 \ 3 \ 4 \ 3 \ 2 \ 3 \ 2 \ 3 \ 4 \ 5 \ 4 \ 3 \ \dots \\
\downarrow \\
\bar{w} = 2 \ 3 \ 4 \ 1 \ 4 \ 5 \ 2 \ 1 \ 4 \ 1 \ 4 \ 5 \ 6 \ 3 \ 2 \ \dots
\end{array}$$

Figure 2.8 \bar{w} results for random output bit sequence (for \bar{w} calculation bit 0 is equivalent to -1)

The changes follow the pattern shown in table 2.1. Based on the output's two consecutive bits ($v[n]$ and $v[n-1]$) there are four different possibilities on how the value of \bar{w} changes. For instance, the sequence of "00", decrease the value of \bar{w} by 1 step size (-1), and the sequence of "01" increase \bar{w} by 3 step size (+3). By using this algorithm, the two paths can be merged into as a single path.

Table 4.1 $\Delta\bar{w}$ for the four possible outcomes based on $v[n]$ and its previous value $v[n-1]$

$v[n-1]$	$v[n]$	$\Delta\bar{w}$
0	0	-1
0	1	3
1	0	-3
1	1	1

Since the feedback signal is subtracted from the analog input, the digital step size variations must be converted to analog as well. This means a reference voltage step size should be chosen, and the step size variations (in this case ± 1 or ± 3) are applied based on that. The process of choosing the reference step size depends on three parameters: the maximum input signal amplitude (A_{IN}), the maximum input signal frequency ($f_{IN_{max}}$), and the oversampling frequency (f_{OS}). In order to make sure that the input signal is tightly tracked by the feedback path, the step size should be selected based on the slope of the input signal (which depends on both A_{IN} and f_{IN}) in a way that the difference between u and \bar{w} is neither too large (low quantization resolution), or too small (slope overload).

Based on this, the relation between reference step size and the three mentioned parameters can be obtained as following

$$\text{Signal's Slope}_{max} = \frac{d}{dt} A_{IN} \sin(2\pi f_{IN} t)_{t=0} = -2\pi A_{IN} f_{IN} \quad (2.5)$$

$$|-2\pi A_{IN} f_{IN}| < \frac{\text{Reference step size}}{\frac{1}{f_{OS}}} \rightarrow \frac{2\pi A_{IN} f_{IN}}{f_{OS}} < \text{Reference step size} \quad (2.6)$$

Expression (2.5) is the maximum slope of a sine wave. In order for the integrating step sizes to follow input signal closely and to prevent slope overload, integrated step size value for each sampling duration ($\frac{1}{f_{OS}}$) should be bigger than the maximum slope of signal. This way the lower bound for the reference step size is defined (2.6). The upper bound for the reference step size value is also set in a way that it is close to target resolution LSB value. Through result of test simulations, to obtain desired SNR from the modulator, the upper bound value shouldn't be more than 1.5 times of the lower bound value.

2.3 MATLAB Simulation Results

The described system was implemented in MATLAB and its efficacy in improving the system's energy efficiency was investigated. In this section, first the MATLAB-based functional implementation of the proposed mixed-signal front is described and simulation results showing its second-order noise-shaping behavior are presented. Next, the functional implementation of the presented adaptive resolution scheme is presented and its effectiveness in improving energy efficiency through loss less data compression is demonstrated using recordings from the CHB-MIT scalp EEG database as test signals [35].

In this system, we picked the two target resolutions to be 8-bit for high resolution recording and 4-bit for low resolution. As discussed previously, these are arbitrary values and the proposed adaptive resolution scheme could be used with more levels of resolution, each set to an arbitrary number. The SQNR of a generic $\Delta\Sigma$ -ADC with order L was shown in (2.1). In addition, the general SQNR for any type of ADC with N-bit resolution is:

$$SQNR = 6.02N + 1.76 \quad (2.7)$$

Using (2.1), and (2.7), and considering an L=2, the oversampling frequencies f_{oslow} , and f_{oshigh} required for the target resolutions are obtained as

$$\begin{array}{l} \text{Target Resolutions} \left\{ \begin{array}{l} \text{8-bit Resolution} \xrightarrow{(2.6)} SQNR=50 \xrightarrow{(2.5)} OSR_{high} = 32 \quad (2.8) \\ \text{4-bit Resolution} \xrightarrow{(2.6)} SQNR=25 \xrightarrow{(2.5)} OSR_{low} = 8 \quad (2.9) \end{array} \right. \end{array}$$

Considering that the maximum input signal's frequency (f_{IN}) is 500Hz, both f_{oslow} , and f_{oshigh} can be calculated with respect to the obtained OSRs.

$$f_{oshigh} = OSR \times 2 \times f_{IN} = 32kHz \quad (2.10)$$

$$f_{oslow} = OSR \times 2 \times f_{IN} = 8kHz \quad (2.11)$$

Figure 2.9 depicts the frequency response of the proposed mixed-signal front-end that is tested with the calculated oversampling frequencies (input signal at 60Hz). The figure on the left shows the result for the case when the system is operating with high resolution. i.e. 8-bit mode, and the figure on the right shows the case that the system operates in the low resolution mode, i.e. 4-bit. As shown, the system shows a 2nd order noise shaping, where noise is pushed to the out-of-band

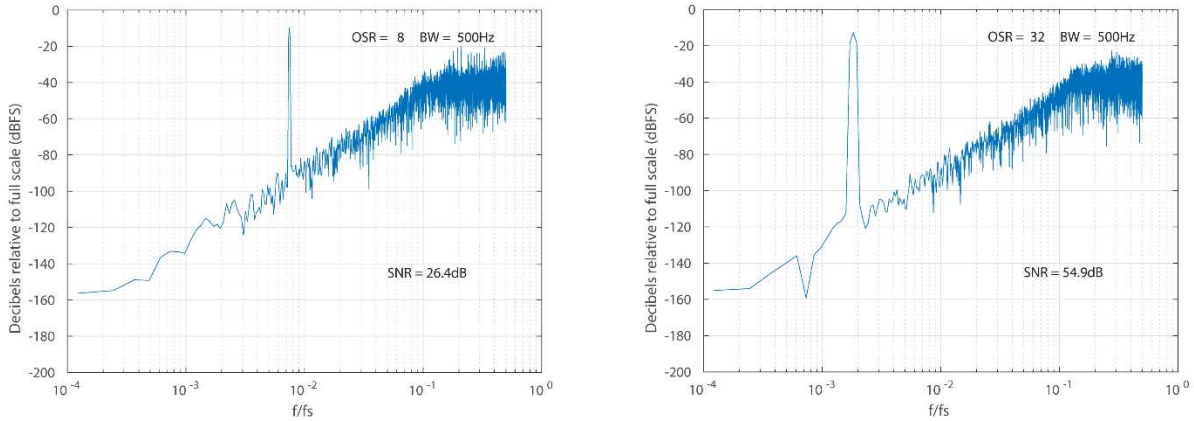


Figure 2.9 Frequency response of the proposed mixed signal front-end in MATLAB, for 4-bit and 8-bit resolution from left to right

frequencies by a slope of 40dB/decade. Also, the calculated SQNR, for each resolution, corresponds to the calculated SQNR in (2.8) and (2.9).

The result of MATLAB simulations of the mixed-signal front-end and digital back-end is shown in Figure 2.10. The blue signal is the raw EEG fed to the system, and the green signal is the output of the presented ADC, sampled with two different resolutions. As shown, when signal's amplitude is higher than a certain threshold level, green waveform closely tracks the original due to the employed high resolution. In contrast, when amplitude is lower than threshold, green waveform coarsely follows the signal due to the lower clock frequency utilized by the modulator. In addition, the red waveform depicts the baseline calculator result which is updated regularly. From the plot, it can be seen DC value is around 0V, meaning ADC's output is also kept around the baseline.

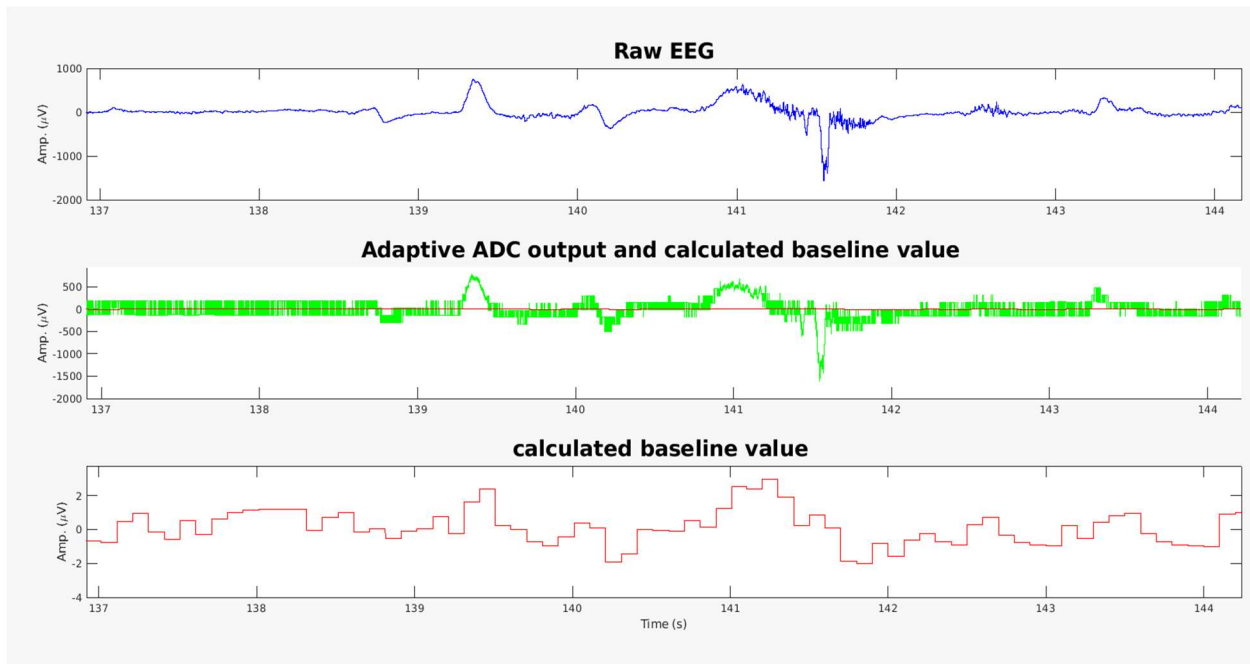


Figure 2.10 EEG signal (top), the reconstructed output with adaptive resolutions (middle), and the automatically calculated baseline (bottom).

In order to determine the efficiency of the method, a set of EEG recordings from CHB-MIT database, each containing a seizure event, are fed to the system. We expect not to miss any of the seizure events as they all are counted as high-activity events, hence, must be recorded with high resolution.

Figure 2.11 depicts the result of digitization (middle) of 10 minutes raw EEG signal (top), with two different resolutions (bottom). In the bottom plot, “0” corresponds to time periods signal was sampled in low resolution, and “1” corresponds to when it was sampled in high resolution. The seizure episodes are marked on the plots. When the signal is in low activity region, meaning it’s lower than threshold level, it is sampled with low resolution, whereas, when the signal is out of this region, it is sampled with high resolution

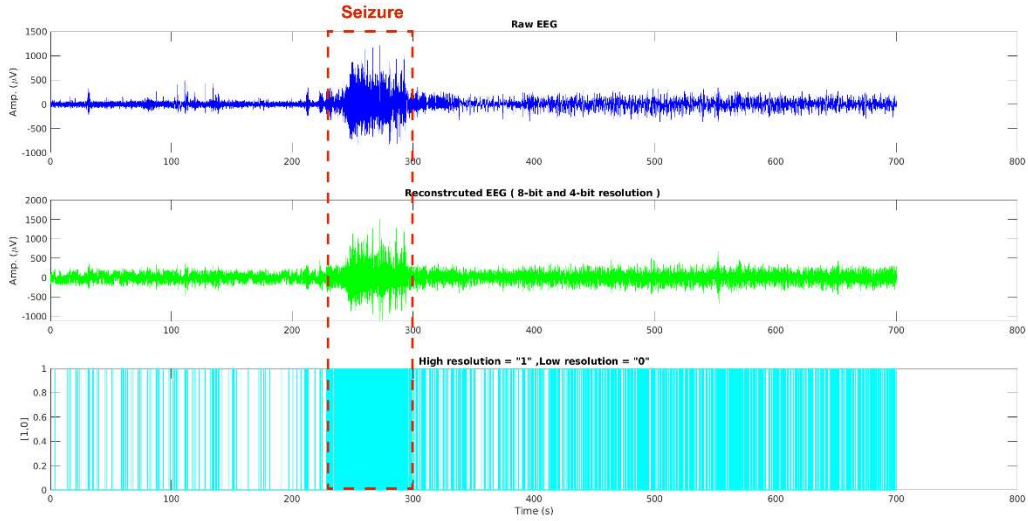


Figure 2.11 An example of system’s response to an EEG signal with seizure

Table 2.2 presents the total value of “1” s and “0” s as well as the percentage of their occurrence during seizures and non-seizure periods of the signal in Figure 2.11. As shown, the test signal was sampled with high resolution (8-bit resolution) for only 20% of the total time, and for the rest, the system operated in low-resolution mode. In addition, for more than half of the time high resolution sampling occurred during the seizure. Also for more than 90% of the total low resolution sampling durations, signal had no seizure. This shows the proposed model behaves as expected, i.e., system’s sampling resolution is adaptive to events occurrence.

Table 2.2 Statistical summary of the simulation results presented in Figure 2.11.

Total % in high resolution sampling mode = 20%		Total % in low resolution sampling mode = 80%	
% in Seizure	% in Non-Seizure	% in Seizure	% in Non-Seizure
53%	47%	6%	94%

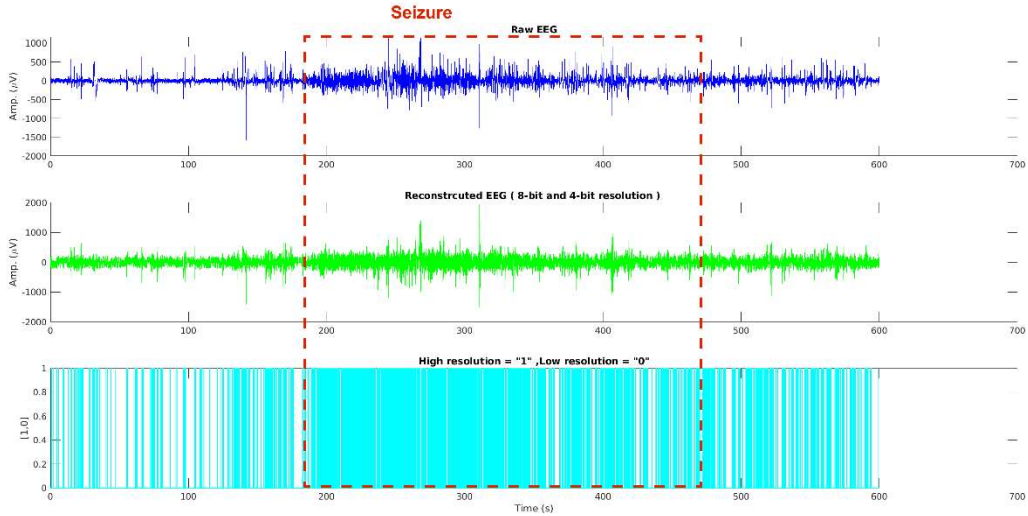


Figure 2.12 An example of the proposed system’s response to an EEG signal containing a seizure episode.

Figure 2.12 shows system’s output for another EEG test signal containing another seizure. Similar to the previous result, it can be seen in Table 2.3, more than half of the high resolution mode is happening within the seizure region, and in total for more than 70% of the total time the system is operating in low resolution mode.

Table 2.3 Statistical summary of the simulation results presented in Figure 2.12.

Total % in high resolution sampling mode = 26%		Total % in low resolution sampling mode = 74%	
% in Seizure	% in Non-Seizure	% in Seizure	% in Non-Seizure
84%	16%	40%	60%

It should be pointed out that the dynamic thresholding technique does not imply seizure detection is an inherent feature of the system. Due to patient-to-patient variations as well as temporal variations, seizure detection is a complicated task often requiring complex data-driven algorithms [59][60]. In the presented system, the suggested technique is capable of determining high-activity events that could be an artifact, noise, interference, or a seizure, and adjusting

resolution of the system accordingly. Seizures are only selected as a show case for neurologically-relevant events.

Lastly, in order to evaluate the power efficiency of this technique, the system was tested for a total of 16 EEG epochs, each containing high activity seizures. To obtain a power efficiency factor for the saved power in wireless transmission, power consumption was calculated for two scenarios of with and without the proposed adaptive resolution.

By calculating the total power consumed for each scenario, it can be seen that power consumption in the transmitter will be nearly halved in the second scenario. The overall power efficiency factor for the tested epochs was calculated as,

$$\eta = \frac{\text{Total power without adaptive resolution}}{\text{Total power with adaptive resolution}} = \frac{17 \times 10^{-6}}{9.75 \times 10^{-6}} = 1.74 \quad (2.12)$$

This is equivalent to an average 43% energy reduction. It must be pointed out that this number could be even higher by (a) reducing the "low resolution" number of bits to 3 or 2, (b) by using more than two levels of resolution, (c) by adjusting the threshold values to make them less sensitive to high-activity episodes.

Chapter 3

Digital Back-End On-Chip Implementation

In the previous chapter, a behavioural system-level implementation of the proposed adaptive-resolution recording channel was described. In this chapter, we will discuss the hardware implementation of the digital back-end, the physical implementation of the design in a standard CMOS sub-micron process, and the developed testbench and measurement setup for the fabricated chip. The digital blocks in this system were first implemented in register transfer level (RTL) using Verilog hardware description language (HDL). Next, using Synopsys Design Compiler [36], they

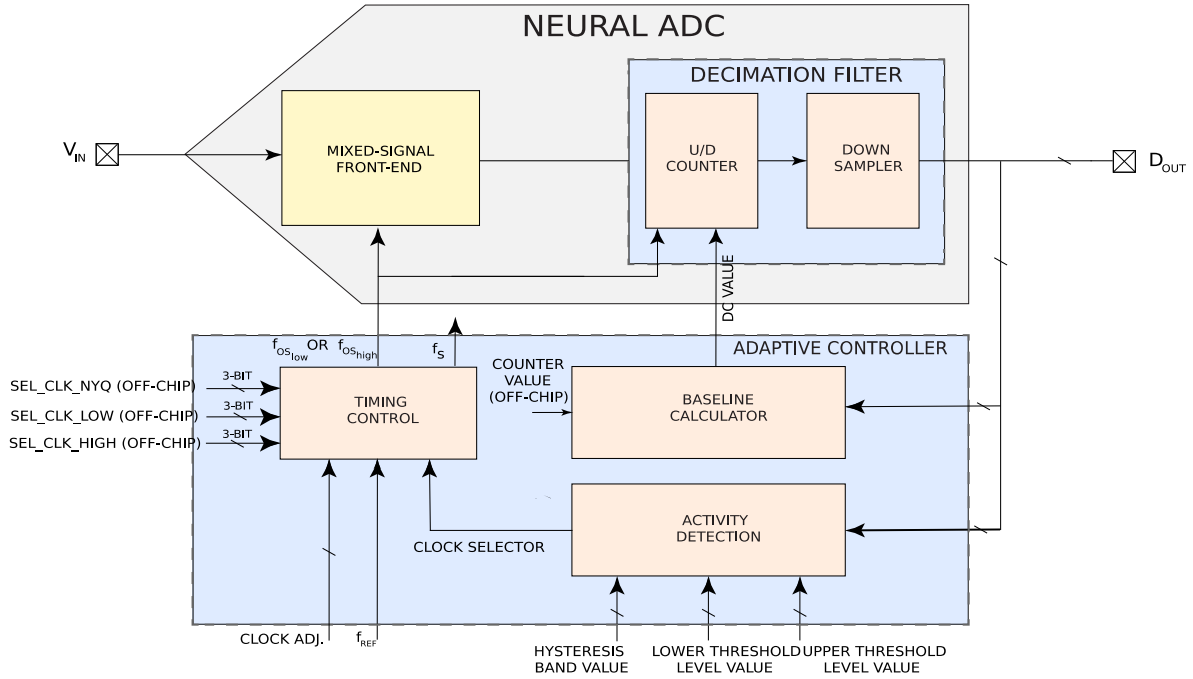


Figure 3.1 Top-level block diagram of the proposed recording channel and the detailed implementation of the digital blocks.

were synthesized and optimized for gate-level implementation, and then were implemented and verified in layout-level using Cadence Innovus, NC-Verilog tools, and eventually entire design was physically verified and signed-off using Cadence Virtuoso tools. In the following sections, the implementation of each of these blocks is described and their simulation results are shown.

3.1 Block-level Implementation

Figure 3.1 depicts the top-level block diagram of in the proposed recording channel as well as the mixed-signal front-end (i.e., the modulator shown in yellow) and the digital back-end (blocks shown in blue). Except for the $\Delta\Sigma$ modulator, all other blocks in this figure are fully digital and are implemented using TSMC 130nm standard CMOS digital cells and with the aid of the kit-associated CAD tools.

3.1.1 Decimation filter

Figure 3.2 shows the internal block diagram of the decimation filter, which is implemented using an 8-bit U/D counter that its output is down sampled to Nyquist-rate frequency (f_s). The output of the $\Delta\Sigma$ modulator which is a 1-bit digital signal is fed to the counter to set the direction of counting (up or down). Also the step size of the counter is set by one of the two control signals selected according to the modulator's clock frequency.

As discussed in chapter 2, in order to reconstruct the input signal through the feedback path of the mixed-signal modulator, an appropriate voltage step size needs to be chosen (the details of the mixed signal front-end implementation is discussed in the next chapter). The voltage step size is chosen according to the desired signal-to-noise ratio (SNR). To reconstruct the signal with two different resolutions in the decimator, the counter's counting step size needs to be set proportional to the voltage step size to achieve correct resolutions.

As an example, for 8 and 4-bit resolutions, if the voltage reference sizes are chosen as $20\mu\text{V}$ and $80\mu\text{V}$, the "HIGH RES. COUNTING STEP SIZE" and "LOW RES. COUNTING STEP SIZE" values in Figure 3.2, need to be set as $8'b00000001$ and $8'b00000100$, respectively. This means that for the high resolution, that is 8-bit, counter changes its value by the smaller step size, and for the low resolution, counter's step size is 4 times larger ($\frac{80\mu\text{V}}{20\mu\text{V}} = 4$). Later during wireless data transmission, for high resolution sampled data, all 8 bits are sent, and for the low resolution case, only 6 bits are needed to be sent as the two LSBs' value never change. In addition, since the low-resolution mode has 4 bits of accuracy in this example, only 4 bits of the remaining 6 bits

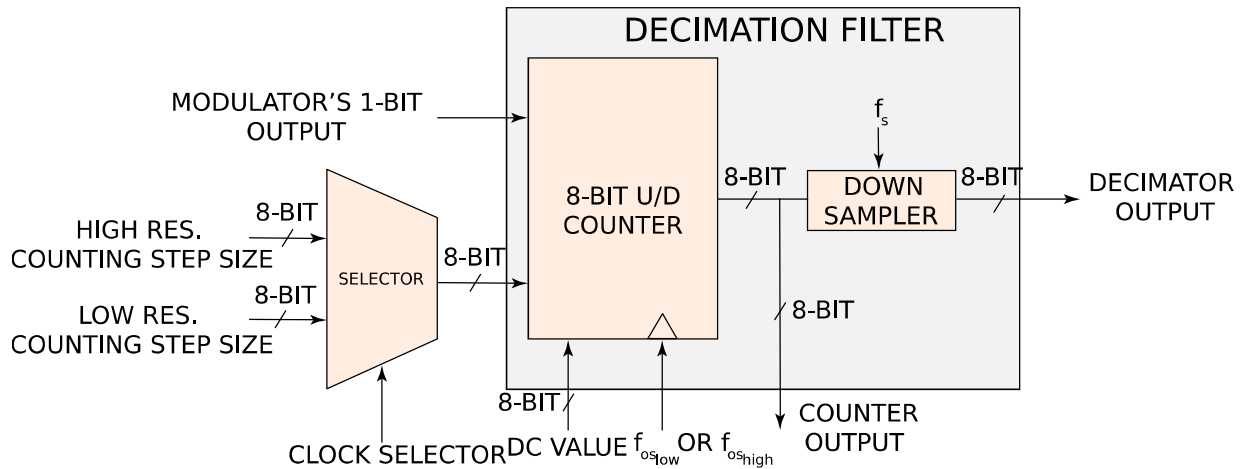


Figure 3.2 Internal block diagram of variable-rate of decimation filter

contain meaningful information, and the two MSBs do not change when operating in low-resolution mode, hence, they will not be transmitted either.

The reduction in the number of bits, as explained in previous chapter, will result in saving power during transmission. It should also be noted that by setting a flag at the beginning of the resolution's change, the receiver will recognize whether the packages are 4-bit or 8-bit information. An automatically adjustable DC value is another input of the decimation filter in Figure 3.2. This input which comes from the baseline calculator unit, is constantly subtracted from the counter's value to always set the reconstructed amplitude at a relatively-fixed DC level.

3.1.2 Activity detection unit

Figure 3.3 shows the operation principle of the activity detection unit and the generated outputs of this unit based on signal's activity. The blue waveform represents the incoming digitized data from the output of the decimator. To determine the level of the activity of the signal, a set of thresholds are defined. The threshold levels as well as the size of hysteresis band are designed to be adjustable. The outcome of the thresholding process is "CLOCK SELECTOR". If signal's

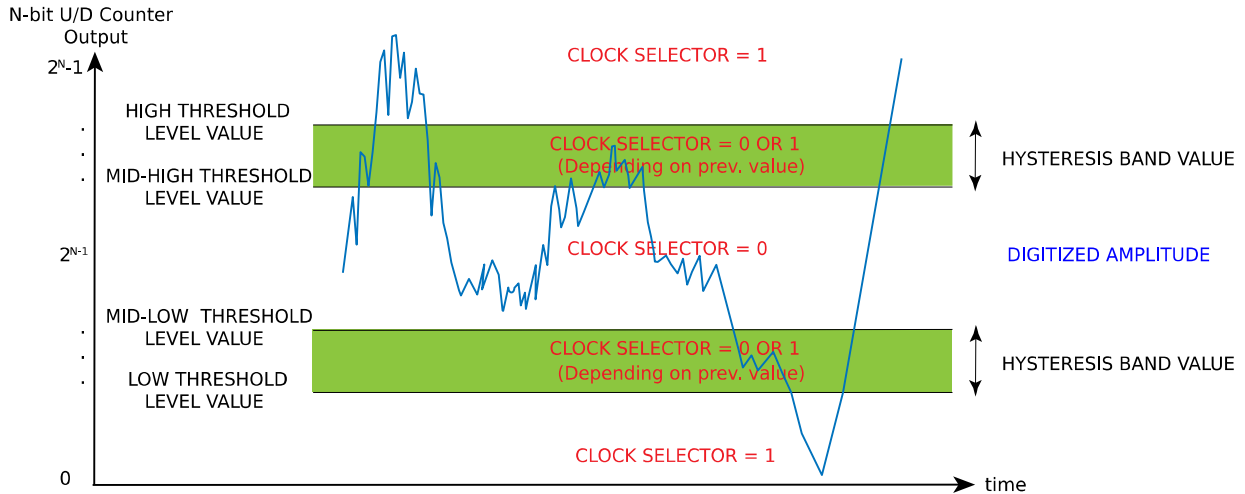


Figure 3.3 Threshold levels and “CLOCK SLECTOR” flag value versus signal magnitude variations.

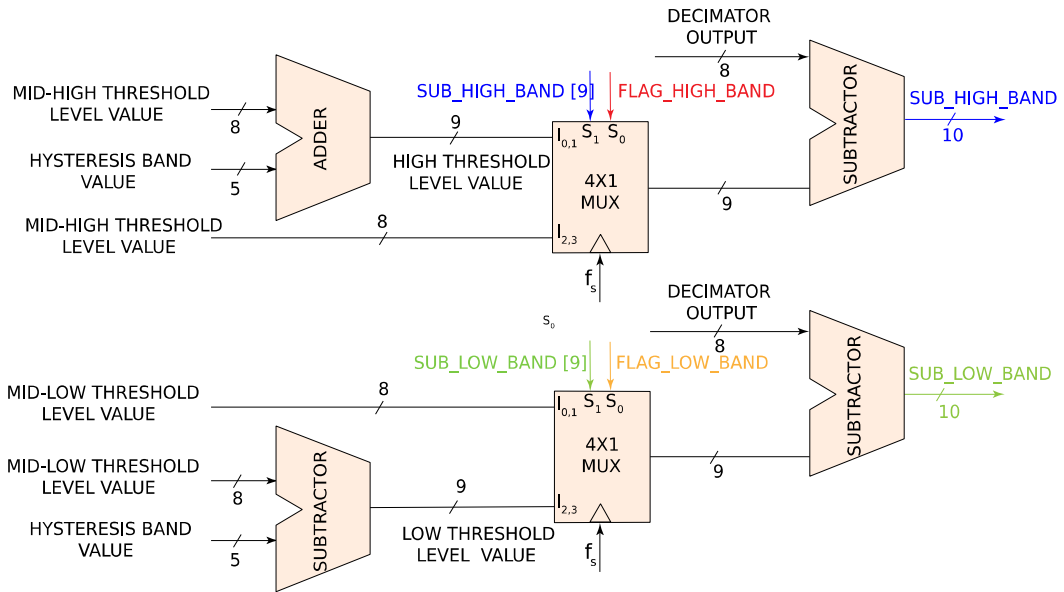
magnitude becomes higher than the highest threshold level (i.e. “HIGH THRESHOLD LEVEL”) or lower than the lowest threshold level (i.e. “LOW THRESHOLD LEVEL”), signal is determined to have high activity, and therefore the “CLOCK SELECTOR” flag will be set to 1 which will change the system’s clock frequency to the high oversampling rate ($f_{OS_{high}}$), resulting in high-resolution quantization. Similarly, if the signal’s amplitude falls between “MID-HIGH THRESHOLD LEVEL” and “MID-LOW THRESHOLD LEVEL” threshold levels, signal is determined to have low activity, resulting in “CLOCK SELECTOR” to be 0, and consequently, operating with $f_{OS_{low}}$ that leads to low-resolution quantization. The hysteresis bands which are the green areas in Figure 3.3, are meant to prevent unwanted oversampling frequency changes for when the signal has very-short-time jumps in its amplitude which don’t really indicate activity change. When signal is in the hysteresis band regions, the “CLOCK SELECTOR” flag will be maintaining its value.

Figure 3.4 shows the internal block diagram of the activity detection unit. The main inputs of this unit are the two threshold levels and the hysteresis band value, all adjustable by the user.

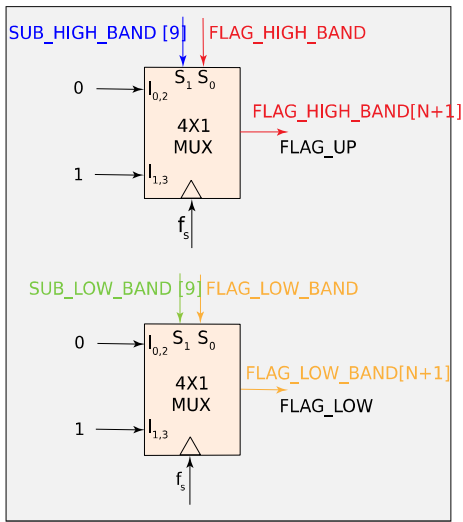
Figure 3.4(a) shows through subtraction/addition, the mid threshold levels are created from the two threshold levels and the hysteresis band value. In order to determine signal's amplitude (i.e. "DECIMATOR OUTPUT") with respect to the threshold levels, 2 main flags called "SUB_HIGH_BAND[9]" and "FLAG_HIGH_BAND" are introduced. Flag "SUB_HIGH_BAND[9]" is the sign bit of subtraction's result of set threshold level (second input of subtractor, which is either set to "HIGH_THRESHOLD_LEVEL" or "MID-HIGH_THRESHOLD_LEVEL" through the multiplexer) and signal's amplitude. Flag "FLAG_HIGH_BAND" indicates where signal stands with respect to the set threshold level. If signal's amplitude is lower than the set threshold level, this flag is 0 and if higher, it is 1. Figure 3.5 and Figure 3.4(b) depicts how this flag is set.

Figure 3.5, depicts how the blue arrow, which indicates the signal position, changes these two flags. As shown, if the set threshold level (indicated in red) is the "HIGH_THRESHOLD_LEVEL", and the signal instantaneous magnitude is below this level, flags will become as "SUB_HIGH_BAND[9]" =1, and "FLAG_HIGH_BAND"=0. Once signal goes above the "HIGH_THRESHOLD_LEVEL", the two flags will change to "SUB_HIGH_BAND[9]" =0, and "FLAG_HIGH_BAND"=1, and consequently, the new set threshold level becomes the "MID-HIGH_THRESHOLD_LEVEL".

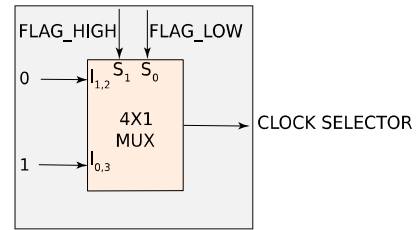
The two flags and the set threshold level will remain the same until the signal's magnitude reduces to lower than this level, when the set threshold level is updated again to "HIGH_THRESHOLD_LEVEL" and flag "SUB_HIGH_BAND[9]" is changed to 1, and flag "FLAG_HIGH_BAND" to 0.



(a)



(b)



(c)

Figure 3.4 Simplified RTL implementation of the activity detection unit: (a) Setting mid-range threshold levels and signal position with respect to the high/low thresholds (b) determining the new value of flag “FLAG_LOW_BAND” and “FLAG_HIGH_BAND” (c) determining “CLOCK_SELECTOR” value.

All the above logic is simultaneously done on the signal instantaneous magnitude and the low threshold levels. The result of both flags “FLAG_HIGH_BAND” and “FLAG_LOW_BAND” is used to evaluate the signal’s position with respect to different regions indicated in Figure 3.3, and to set the “CLOCK_SELECTOR” flag accordingly as shown in Figure 3.4(c).

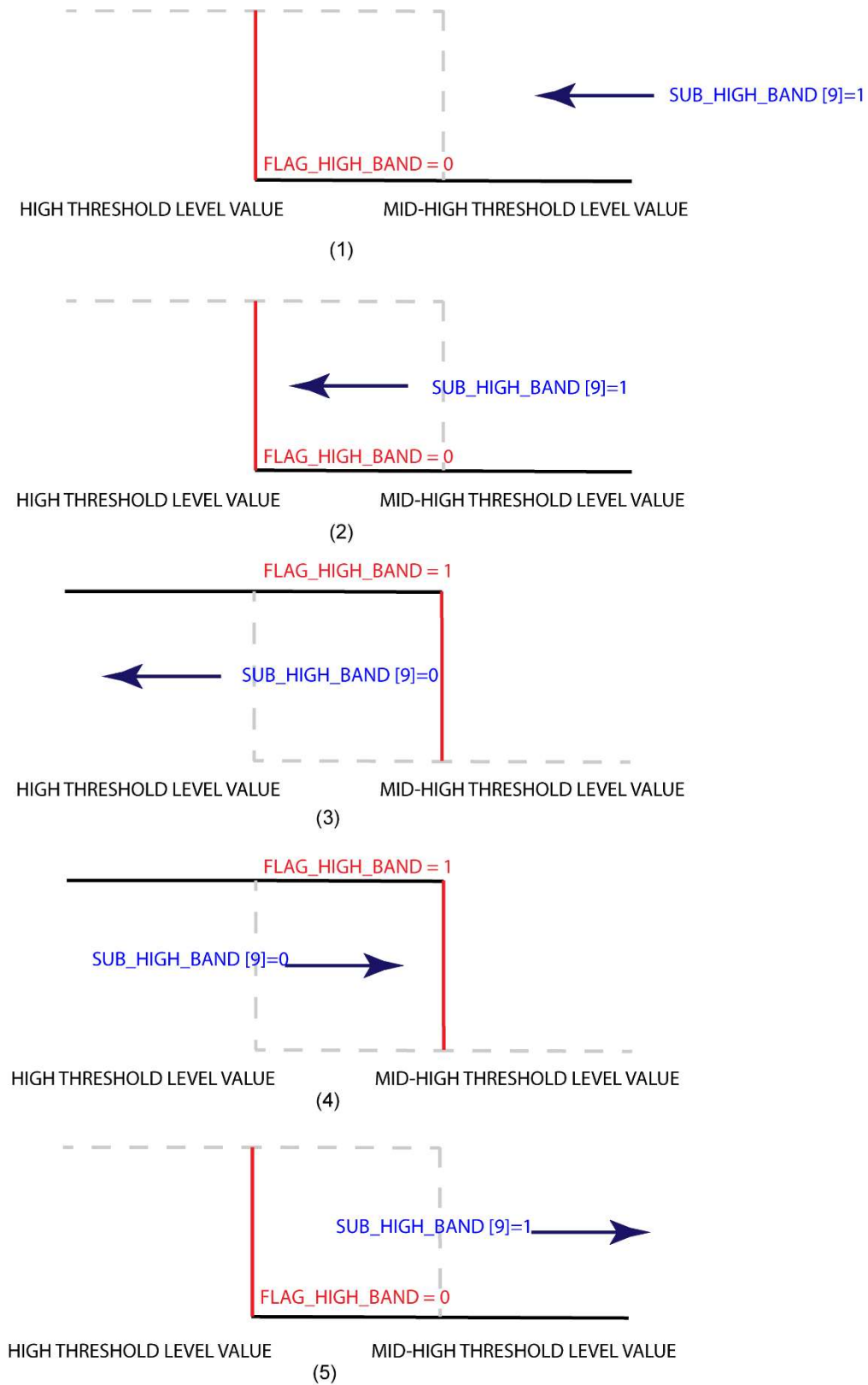


Figure 3.5 Flags behavior for the activity detection unit with respect to threshold levels, figure only depicts for high threshold levels, same behavior for low threshold levels

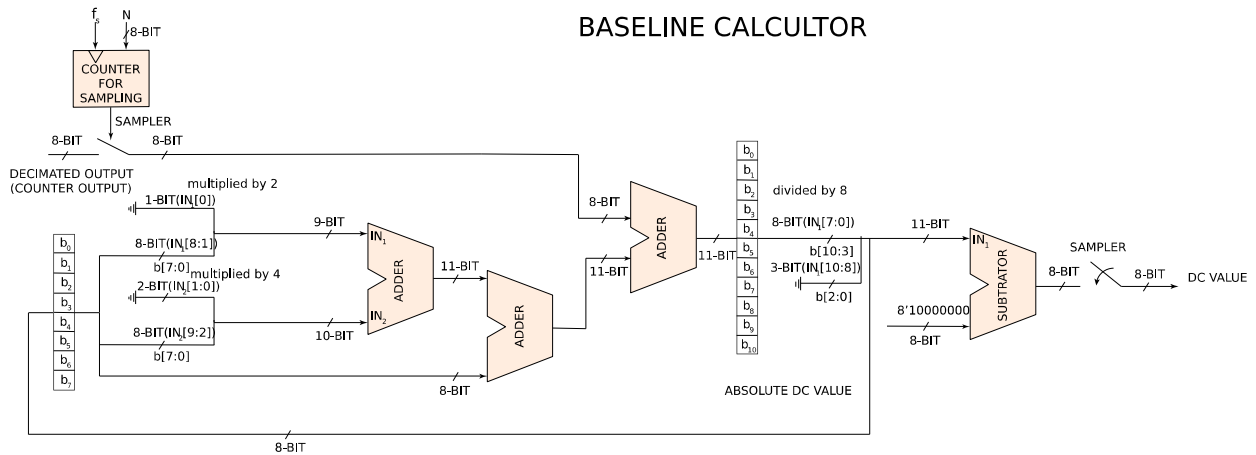


Figure 3.6 Internal block diagram of the baseline calculator unit

3.1.3 Baseline calculator unit

Figure 3.6 shows the internal block diagram of the baseline calculator unit. As discussed in Chapter 2, the real-time moving DC value of the time-domain signal amplitude is calculated over a programmable period and is subtracted from the signal's instantaneous magnitude before being fed to the activity detector. This will allow for detecting the actual short-time high-activity events independent of the signal's slowly-varying DC level. As shown in Figure 3.6, the decimator's output (i.e., the down-sampled counter's output) is the input of the baseline calculator unit, which is sampled every N Nyquist cycles. This sample is used to update the baseline value that is stored in a register using a weighted averaging process described below. "Absolute DC value" is the value of baseline before it is subtracted from 128 in order to create "DC Value" with respect to decimator's original baseline.

The value of N should be set in a way that the period between each amplitude reading for updating the DC value is neither too long nor too short. If the reading period is much shorter than the duration of a typical high activity period, there will be a chance that the large amplitude of the

high-activity event is mistaken as a change in the signal's DC level, even resulting in missing an event due to the raised DC level. Also if the duration between each calculation time is too long, it will take too long for the DC level to be updated, potentially resulting in either missing some of the events or detecting an entirely idle period as a high-activity period because of their high DC value, simply because the DC level change is captured too late.

In addition to optimizing the reading period, incorporating the sampled data into previously-calculated DC value is also of critical importance. To update the DC value based on a new sample, the current DC value, which is the result of all samples from the beginning of the recording until this moment, should have a much higher weight than the new sample. A small weight means that the DC could change dramatically every time a new sample is taken, especially if the samples falls exactly on a signal peak or trough. A very large weight will have the same effect as a very large N , i.e., results in a very slow DC change that could lead to the above-mentioned mis-detections. In this work, we used a 7-to-1 weighting based on our simulation results on pre-recorded offline EEG data. As shown in Figure 3.6, the value of N is programmable and can be adjusted post-fabrication.

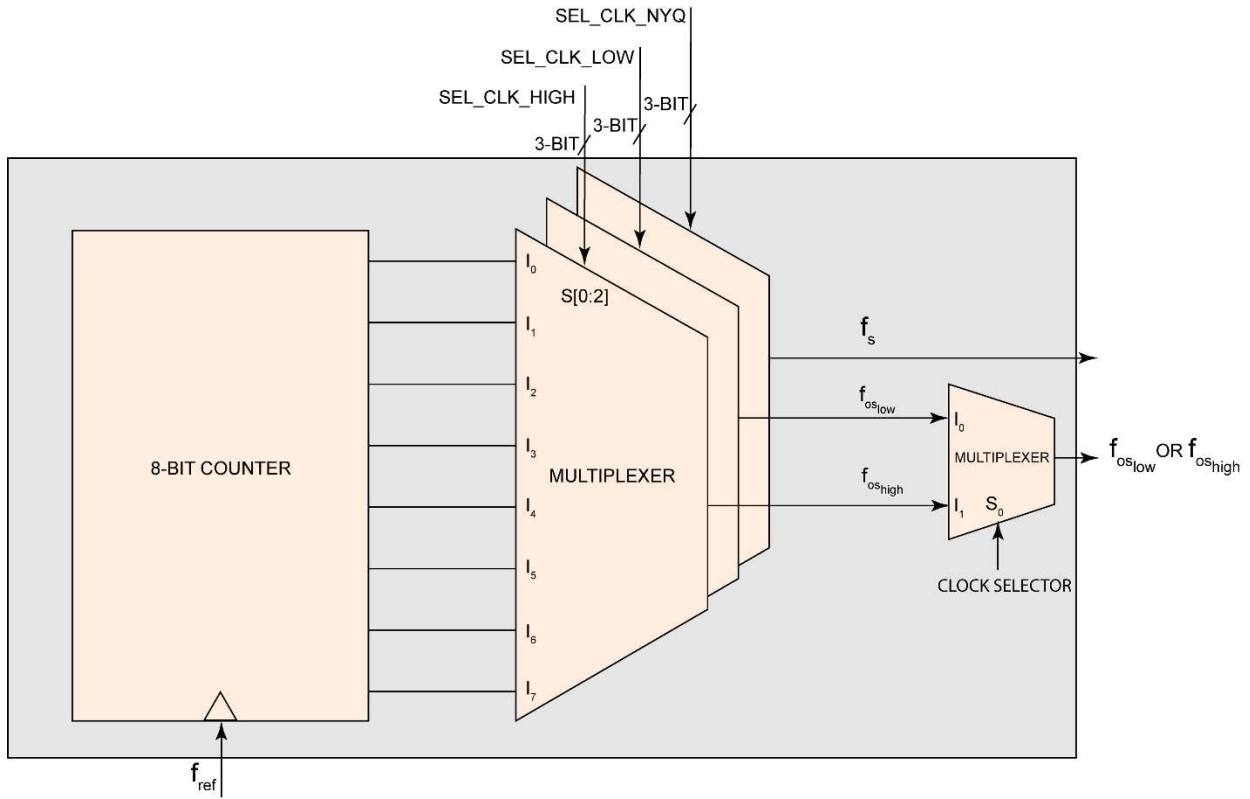


Figure 3.7 Logic of clock generator & selector unit

3.1.4 Clock generator and selector unit

Figure 3.7 shows the clock generator and selector unit. Using the output of the 8-bit counter, eight frequencies are available, which are binary divisions of the reference clock frequency (f_{ref}) that is adjustable by the user. As the highest required oversampling clock frequency is 64kHz, the reference clock frequency fed to the system was set to 128kHz. The output of counter is fed to three multiplexer, which by the 3-bit select signals, i.e. “SEL_CLK_HIGH”, “SEL_CLK_LOW”, and “SEL_CLK_NYQ” from off-chip, clock frequencies f_{os_high} , f_{os_low} , and f_s are set respectively. Also as shown, the “CLOCK SELECTOR” flag, controlled by the activity detection output, selects the system’s oversampling frequency between two signals f_{os_high} , and f_{os_low} .

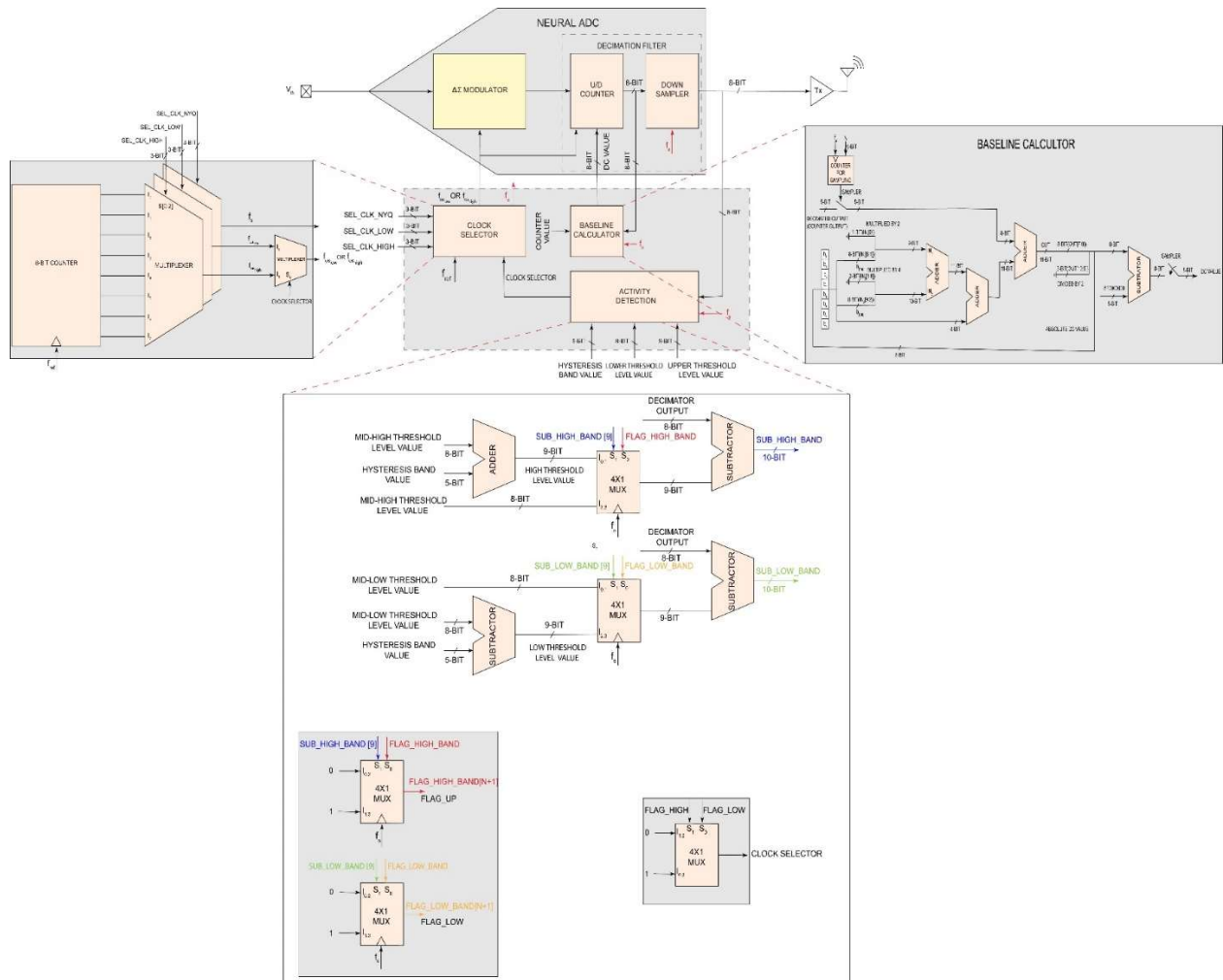


Figure 3.8 Overall block diagram of system

3.2 Overall Back-end Integration and Implementation

Figure 3.8 shows the overall block diagram of the implemented system. Table 3.1, shows system’s input values (e.g. threshold levels, hysteresis band, clock frequencies) for a test example. The system’s performance was validated using both gate-level and layout-level netlist. The following simulation results are obtained from layout-level netlist using Cadence NC-Verilog tools.

Table 3.1 Digital back-end input values

INPUTS	VALUES
SEL_CLK_NYQ	4'b0110
SEL_CLK_HIGH	4'b0001
SEL_CLK_LOW	4'b0011
HYSTERESIS BAND VALUE	5'b001000
MID-HIGH THRESHOLD LEVEL VALUE	8'b10001010
MID-LOW THRESHOLD LEVEL VALUE	8'b01110110
HIGH RES. COUNTING STEP SIZE	8'b00000001
LOW RES. COUNTING STEP SIZE	8'b00000111
DC COUNTER VALUE	8'b00010011

In this test, system's input clock frequency, f_{ref} (Figure 3.8) is set at 128kHz , and by appropriately setting the clock selector flags (as shown in Table 3.1), f_s is set to 1kHz, $f_{OS_{high}}$ at 32kHz and $f_{OS_{low}}$ at 8kHz. Figure 3.9 is the simulation result for the clock generator block. Waveform f_{ref} which is the reference clock is set at 128kHz. Waveform f_s is the nyquist sampling rate and is 1kHz. The "CLOCK SELECTOR" flag which comes from the activity detection unit selects the oversampling frequency. When this flag is 1, high oversampling frequency , $f_{OS_{high}}$ (set at 32kHz) is the selected oversampling frequency of system , and when the flag is 0, low sampling frequency is $f_{OS_{low}}$ (set at 8kHz).

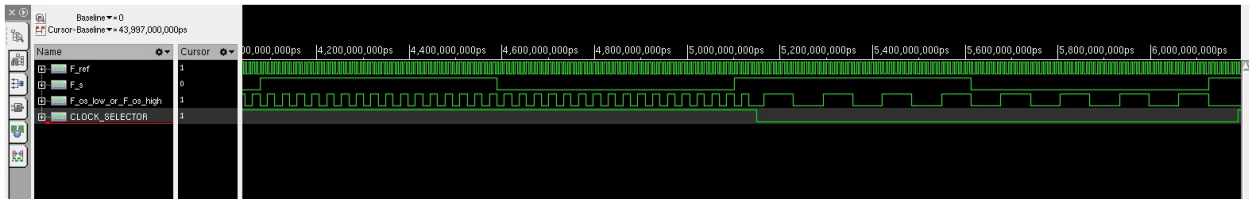
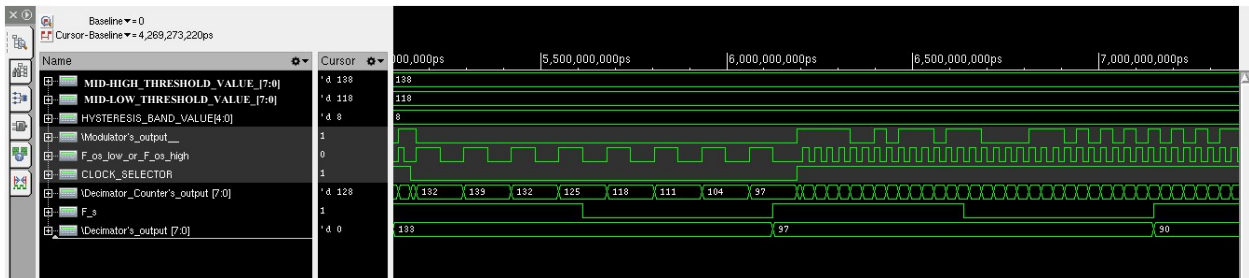
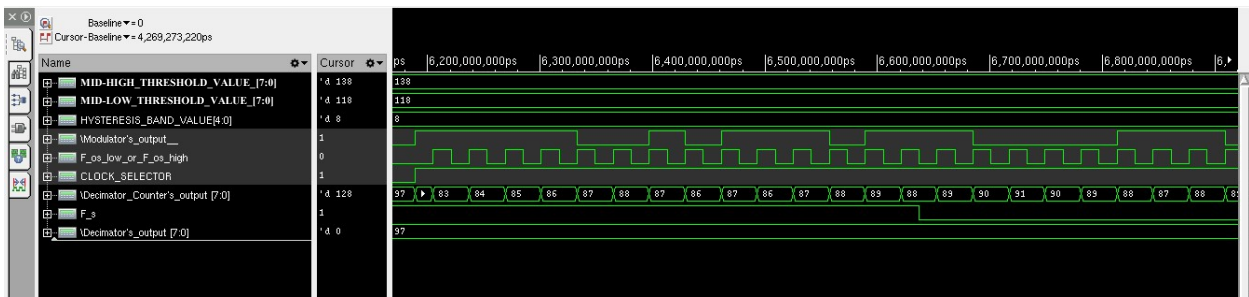


Figure 3.9 Simulation results for clock generator unit



(a)



(b)

Figure 3.10 Simulation results for activity detection and decimation filter unit

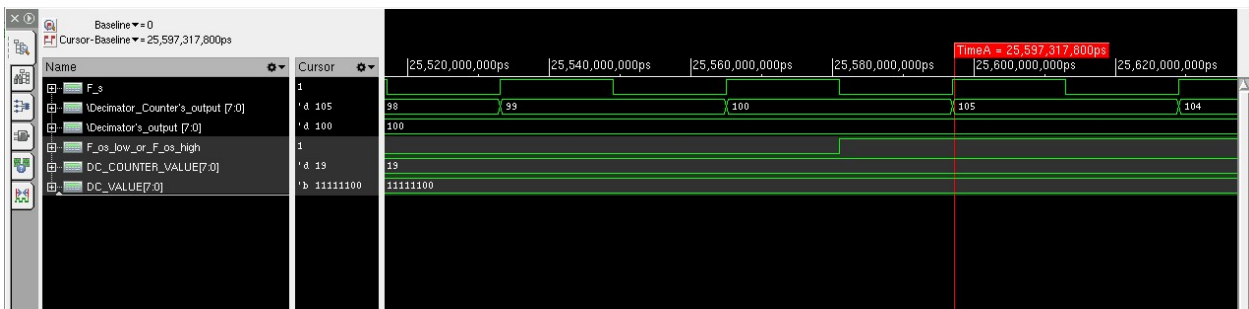


Figure 3.11 Simulation results for baseline calculator unit

Figure 3.10 shows the simulation result of decimation filter and activity detection unit. It can be seen that the modulator's 1-bit output (waveform Modulator's_output) is read at its corresponding oversampling frequency ($f_{OS_{high}}$ or $f_{OS_{low}}$). As explained in section 3.1.1, decimator's counter's value changes according to the low or high resolution step sizes. When system is in the low resolution mode ("CLOCK_SELECTOR"=0), it can be seen in Figure 3.10(a) and (b), counter's output (waveform Decimator_counter's_output) changes by 7 step sizes (as set in "LOW RES. COUNTING STEP SIZE"), and when in the high resolution mode, counter changes by 1 step size (as set in "HIGH RES. COUNTING STEP SIZE"). Counter's output is then down sampled and its value is read at f_s frequency (waveform Decimator's_output). Decimator's output is compared to the threshold levels. Since 138 and 118 are mid-high and mid-low threshold level values (for the scale of 0-255 with 128 being the baseline) and 8 is the value of hysteresis band, high and low threshold levels will be 146 and 110 respectively. It can be seen when decimator's output is 133, oversampling frequency is set low (i.e., choosing $f_{OS_{low}}$). Once decimator's output becomes 97, oversampling frequency is changed to high (i.e., choosing $f_{OS_{high}}$), since signal goes to the high activity region.

Figure 3.11, depicts the result of simulation for baseline calculator unit. The DC value is calculated every 19 samples (as set in "DC COUNTER VALUE"). Once DC value is calculated, it is subtracted from the counter's output. In the figure, it is shown that the calculated DC value is -4 (i.e., 2's complement 8'b11111100), therefore, the counter's output is shifted up by 4 before being fed to the activity detection unit (decimator's output is 100 and then in the next nyquist clock edge, its value has changed by 5 (105), which 1 is for modulator's bitstream and 4 for calculated DC value).

It should be noted that all the input values, (e.g. threshold levels, hysteresis band, DC counter, etc.) are set according to the target signal, and desired performance. For the case of this simulation by examining EEG data for various patients, it was seen through MATLAB simulations, the set input values give the desired results.

3.3 Physical Layout

The physical layout of the proposed digital back-end was implemented and optimized in Cadence Innovus tools, and chip sign-off process was done using Cadence Virtuoso.

In table 3.2, digital back-end's performance summary and specifications are provided. It should be pointed out that some of the blocks (e.g., counter for clock generation) are shared between channels and the listed total power consumption (i.e., $1.91\mu W$) decreases as the number of channels are increased.

Table 3.2 Digital back-end's specifications

Spec.	
Tech.	TSMC 130nm, 8 metal layers
Supply	1.2V
Max Acceptable Freq.	128kHz
Total Power Consumption	$1.91\mu W @ 128kHz$
Net active Area of All Cells	$92 \times 92\mu m$
Total Number Of Combinational Cells	446
Total Number Of Sequential Cells	110

3.3.1 Power overhead of Digital Back-end

To look into the power overhead added to the design through the addition of the digital back-end, power consumption in a typical implantable BMI without the implemented technique is calculated and through the efficiency factor obtained in chapter 2, the power overhead of the additional blocks compare to overall power saving is shown.

A typical implantable BMI has 64 channels, with a maximum 10kHz signal bandwidth, and an 8-bit resolution. with a transmitter with power consumption of 1nJ/bit [43]:

$$2 \times 10\text{kHz} \times 64 \times 8\text{bits} = 10\text{Mbps} \quad (3.1)$$

$$\frac{1\text{nJ}}{\text{bit}} \times 10\text{Mbps} = 10\text{mW} \quad (3.2)$$

By using the proposed technique (for the case of 8-bit 4-bit resolution) 10mW will reduce by 43%, meaning 5.7mW power is consumed by addition of less than $2\mu\text{W} \times 64 = 0.128\text{mW}$ (if the digital back-end is added to each channel individually) which is considerably lower than the total power saved. Knowing that future direction of this field is towards integrating hundreds/thousands of channels on a single chip, the power benefits of this proposed scheme become increasingly more significant.

3.4 chip Fabrication and Measurement Setup

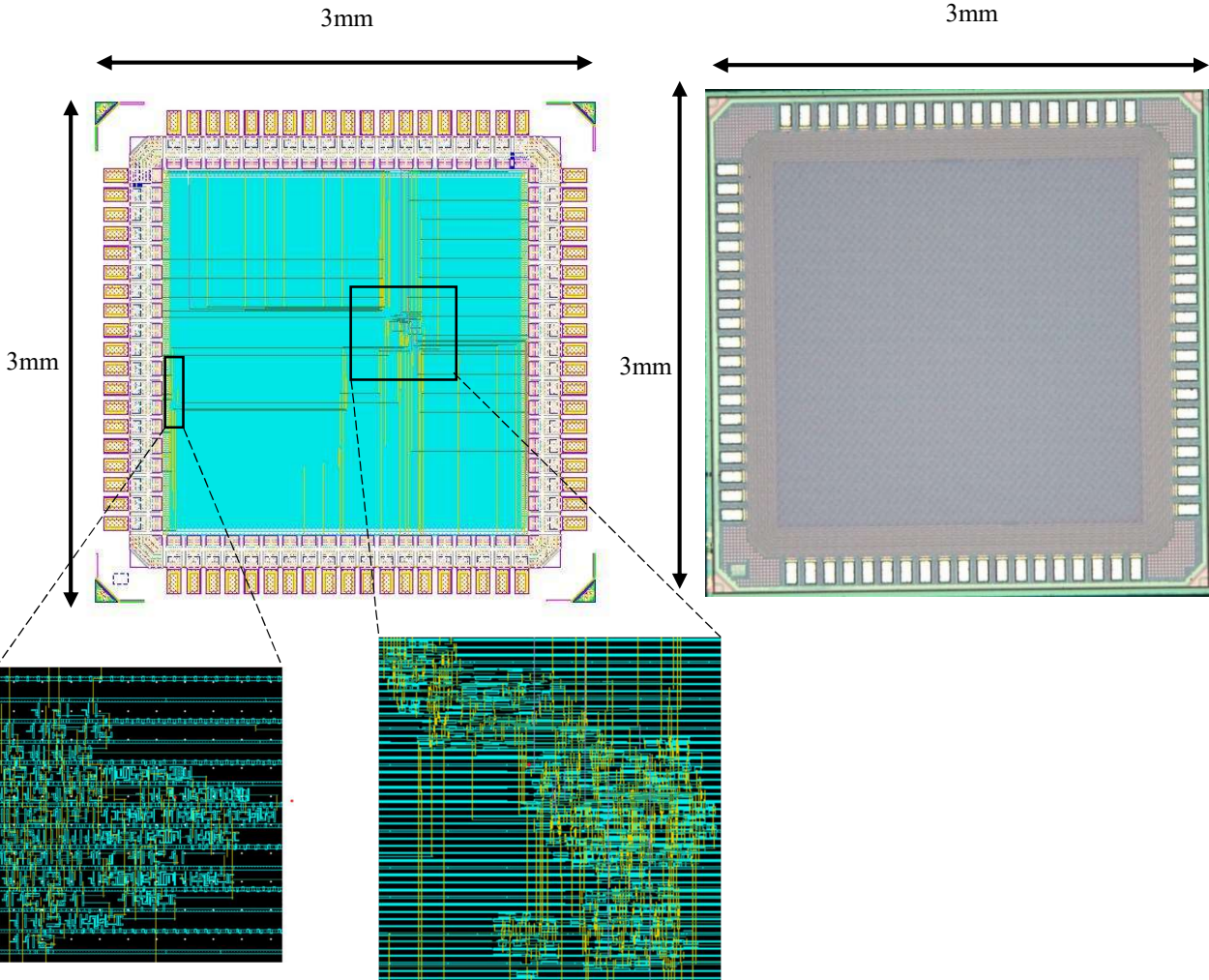


Figure 3.12 Chip's layout and the micrograph

Figure 3.12 shows chip's shows the full-chip physical layout as well as fabricated micrograph. Figure 3.13 depicts the lab set up for chip measurement. The detail annotation and explanation of the measurement set up and PCB are discussed in appendix.A.3.

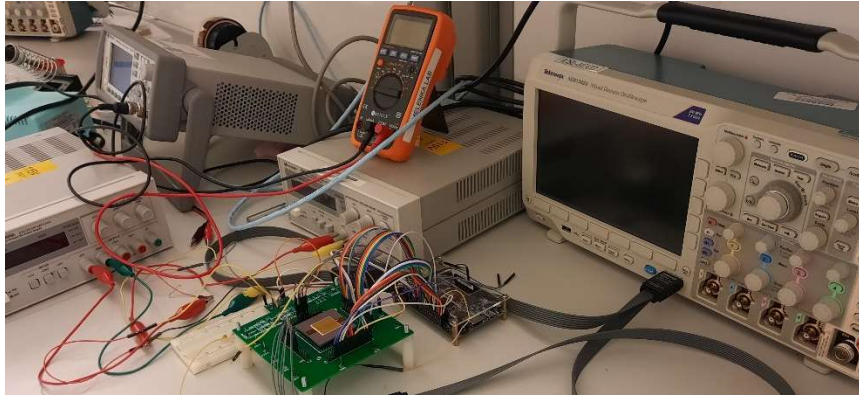


Figure 3.13 Lab set up for chip's measurement test

Chapter 4

Mixed-Signal Front-End Implementation

In this chapter, block-level and transistor-level implementation of the proposed mixed-signal front-end will be presented. The transistor-level implementation and the time- and frequency-domain verifications are done in Cadence Virtuoso tool using TSMC 130nm technology kit. Figure 4.1 shows the top-level block diagram of the proposed recording channel that consists of a mixed-signal neural ADC connected to the digital backend blocks that control the operation of the ADC to make its resolution adaptive to the input signal's level of activity. The implementation of the digital blocks was discussed in Chapter 3. In this chapter, we will discuss the detailed implementation and test results of the mixed-signal front-end circuit. As discussed in chapter 2, the front-end is expected to have an equivalent performance of a 2nd order $\Delta\Sigma$ modulator, i.e. providing 40dB/dec noise shaping. In addition, the digital-to-analog converter (DAC) in the ADC's

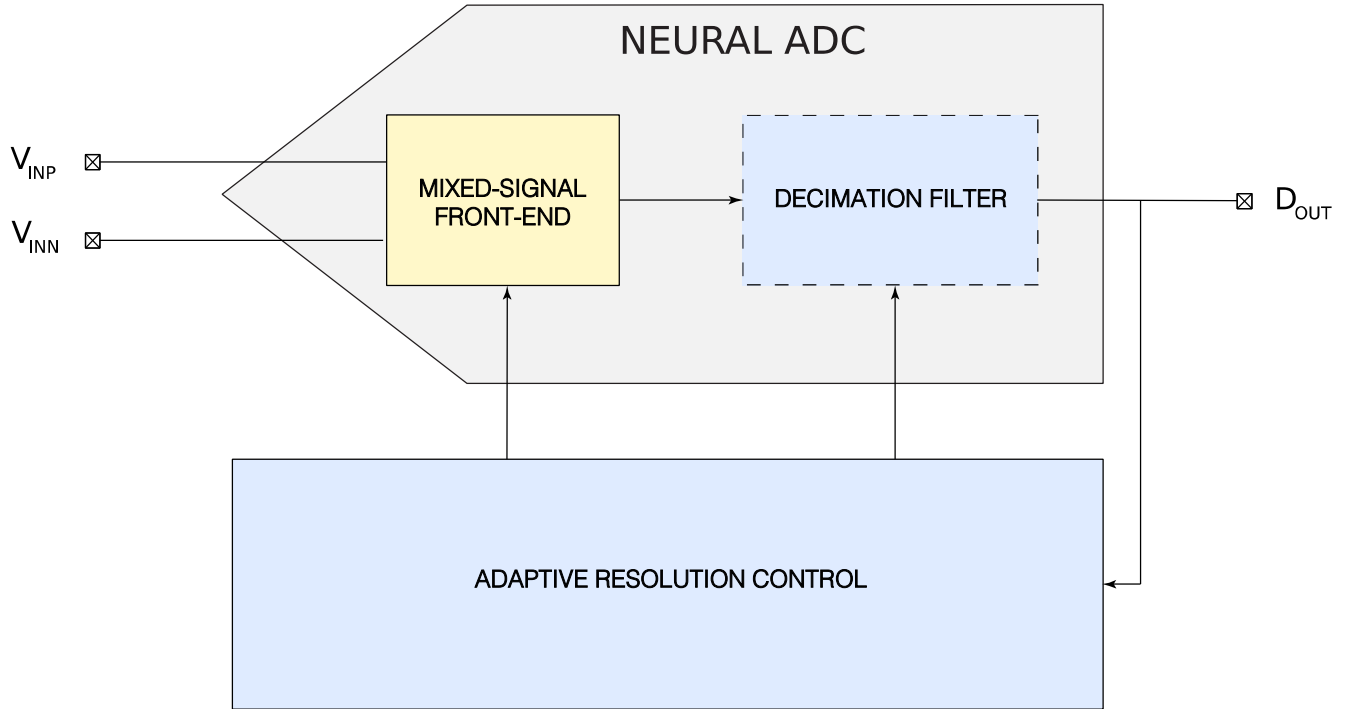


Figure 4.1 Top level block diagram of the presented recording channel.

feedback path of the modulator must be designed such that it can generate an output that closely follows the input signal's low-frequency dynamics, hence removes DC offset and drifts at the input.

Table 4.1 shows the required design specification of the neural recording front-end.

Table 4.1 Specifications of the neural front-end

Spec.	Target
DR(dB)	>50
Input Range (V)	10 μ -1m
Bandwidth (Hz)	500
Input DC drift Cancellation (V)	>50m
IRN($V_{rms(1-500Hz)}$)	< \sim 5 μ
Input Impedance (Ω)	>100M
Power and Area	Min. possible value

In the following sections, the design procedure and the performance characterization of each of these blocks will be discussed.

4.1 Digital-to-analog converter (DAC)

In chapter 2, the principles of the 2nd order $\Delta\Sigma$ modulator and its system-level behavior was discussed. Figure 4.2(a) shows the architecture of a discrete-time (DT) 2nd order $\Delta\Sigma$ modulator, in which the input $u(t)$ is applied to the second summing node and one of the inputs of the first summing node is made equal to 0. As discussed in Chapter2, this configuration results in a high-pass shaping of the STF, which leads to removing unwanted DC offsets/drifts. It was also shown that by varying the feedback DAC's step size according to the modulator output bitstream, the

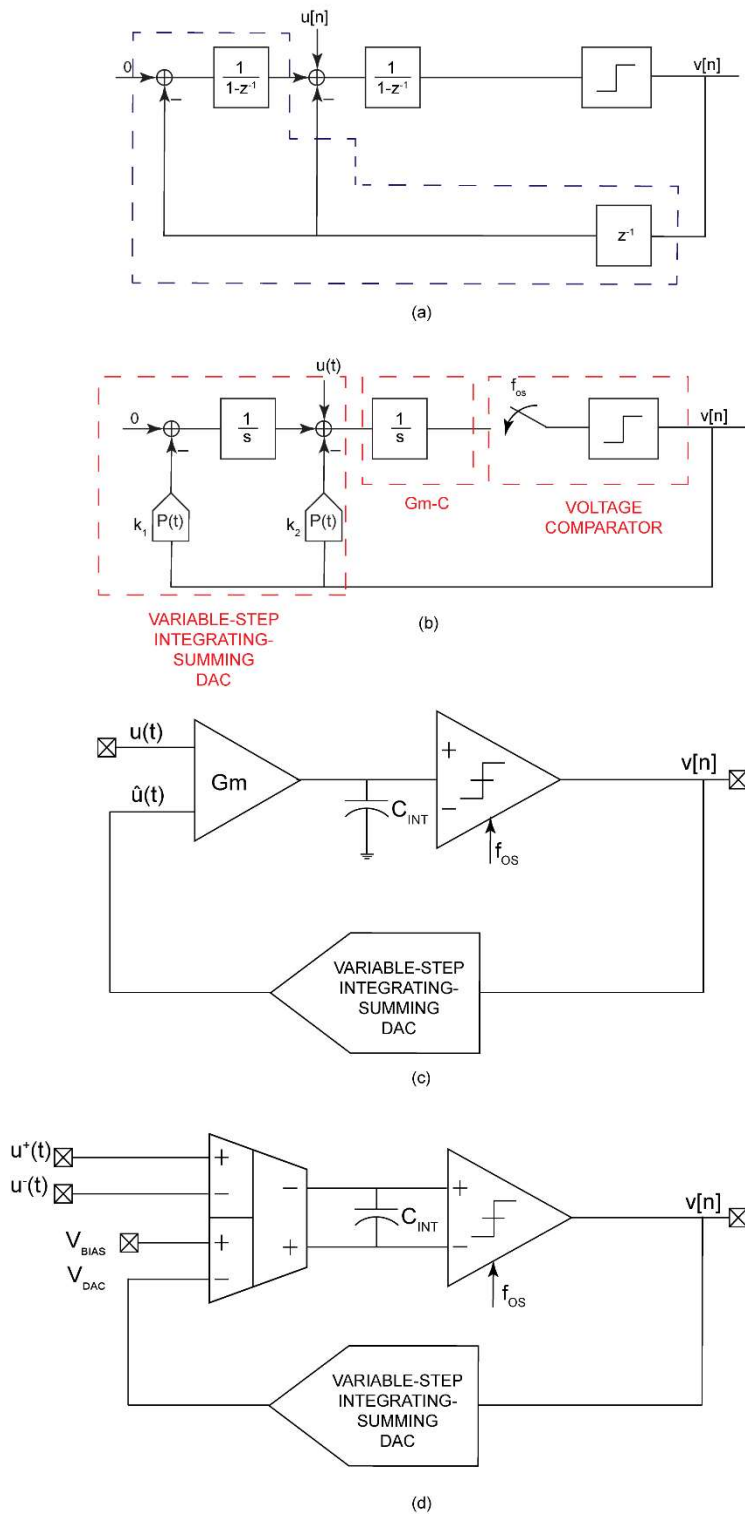


Figure 4.2 (a) Discrete time (DT) and (b) continuous-time (CT) 2nd order $\Delta\Sigma$ modulator block diagram with input arrangements that result in second-order noise shaping and first-order signal shaping (i.e., Δ modulation). Proposed mixed-signal front-end architecture

DAC effectively integrates the output bitstream and adds the result to the output bitstream itself (i.e., has the equivalent functionality of all blocks inside the dashed box of Figure 4.2(a)).

In this design, the modulator is implemented in continuous-time(CT) domain, meaning that the input signal is sampled after being passed through the loop filter (Figure 4.2(b)). This is mainly to take advantage of the inherent anti-aliasing of CT modulators. In terms of the block diagram, the main difference between a standard discrete-time (DT) modulator and its CT counterpart is the DACs with the pulse shape $P(t)$ (the pulse shape function $P(t)$ could be various depending the application) that are introduced in the feedback paths of the CT modulator. The addition of the DACs makes the impulse response of the CT and DT modulator's loop filter different, which ultimately causes difference in system's behavior for the same order DT and CT. In order to maintain the same system behavior, the loop filter of both systems must become the same. Because of this, a "k" coefficient for each DAC is introduced, and by choosing proper values for these coefficients, CT and DT modulators show similar behavior [34]. In addition, with this structure, stability issues introduced in oversampling ADCs are avoided since stability concerns for this type of ADC are more of a concern for orders and quantization levels above two.

The annotations on Figure 4.2 (b) shows how each set of blocks are implemented using actual electronic circuit blocks. Figure 4.2 (c) shows how these blocks are connected together to form the proposed modulator with a single-ended input. Figure 4.2 (d) shows the fully-differential version of the proposed design. The design consists of a differential-difference transconductance amplifier (DDTA) connected to a capacitor (C_{INT}) to form a gm-c integrator stage, followed by a voltage comparator that acts as the 1-bit quantizer, and a variable-step integrating-summing DAC in the feedback path that is supposed to add the output bitstream with its integration and feed the result to one of the DDTA's inputs.

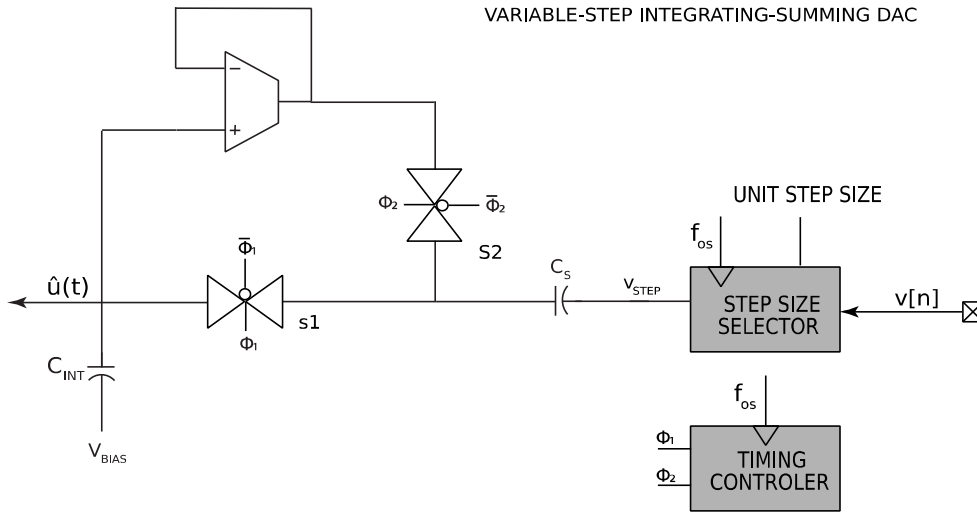


Figure 4.3 Internal architecture of the variable-step integrating-summing DAC

Figure 4.3, shows the internal block diagram of the above-mentioned integrating-summing DAC. The modulator's output, which is a stream of 1s and 0s ($v[n]$) is fed into a voltage level selector unit. In this unit depending on the sequence of the last two bits (as discussed in detail in chapter 2) one of the two voltage step sizes (voltage level = $1 \times$ step size, or voltage level = $3 \times$ step size) is chosen. Depending on the step size, a rising voltage step with a magnitude of (20mV or 60mV) will be created by the voltage level selector, which is shown as V_{step} in the Figure.

When S1 is closed and S2 is open, C_S and C_{INT} will be in series with each other, therefore, they form a capacitive voltage divider that results in changing the node voltage of $\hat{u}(t)$ by $\frac{C_S}{C_S + C_{INT}} \times V_{step}$. This change could be an increase or a decrease, depending on the direction of the step voltage being positive or negative, respectively. By choosing C_S to be much smaller than C_{INT} , the step size voltage could be made arbitrarily small. In this work, for the resolution of 8-bit we targeted a step size of $20\mu\text{V}$, hence chose $C_S = 10\text{fF}$ and $C_{INT} = 10\text{pF}$.

The above-described circuit only works for one cycle. Once the V_{step} is reset to zero to be ready for the next positive/negative step, $\hat{u}(t)$'s voltage changes accordingly with the same $\frac{C_S}{C_S+C_{INT}}$ ratio, unless S1 is open (hence, the reason for having S1). This way, resetting V_{step} will not affect $\hat{u}(t)$'s voltage. However, since the left plate of C_S is floating, any change in V_{step} due to resetting will be copied on the left plate of C_S . This is because C_S 's current ($C_S \frac{dV}{dt}$) is zero, thus dV is zero, hence the voltage across C_S must remain constant. This is a serious issue because when we want to close S1 in the next cycle, the two ends of the switch have different voltages, hence will result in a significant charge to be pulled/pushed to the C_{INT} , hence changing the $\hat{u}(t)$'s voltage and ruining the entire process.

To prevent this, we need to make sure that the $\hat{u}(t)$'s voltage is copied onto the C_S 's left plate, right before S1 is closed. As such, the OTA-based voltage buffer and S2 are added to the circuit to perform the voltage copying.

Finally, the bottom plate of the C_{INT} is connected to the same V_{bias} as the third terminal of the differential difference transconductance amplifier shown in Figure 4.2(d). This will ensure that the low-frequency (i.e., DC offset or drift) difference between the 3rd and 4th inputs of the differential difference stage is equal to that of the 1st and 2nd inputs (that are connected to the electrodes), which will be removed by the differential-difference structure.

To minimize the switches (S1 and S2) non-idealities (e.g., charge injection), the size of transmission gates is minimized. Also, in designing the OTA of this circuit, to have the voltage of the positive and negative inputs as close as possible (less than $5\mu\text{V}$ difference is accepted in this design, considering the targeted resolution), the voltage gain was set to $>60\text{dB}$. Additionally, the unity gain bandwidth (UGBW) of the open-loop amplifier should be high enough so that when

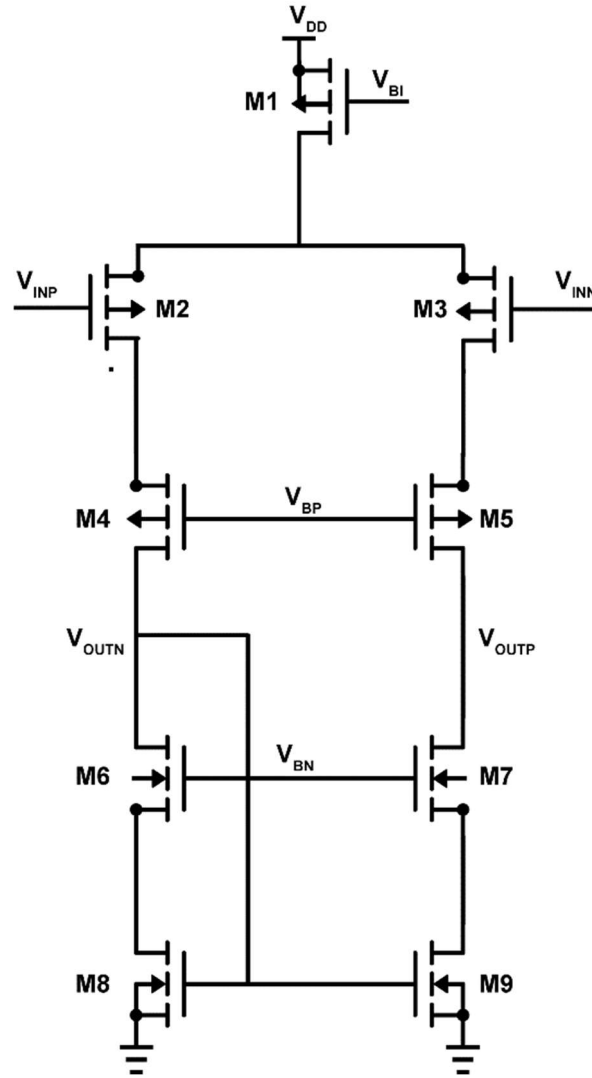


Figure 4.4 Telescopic OTA used in the DAC circuit

used in buffer configuration, the OTA's output is settled before S2 is turned off (less than half of the highest oversampling period, i.e., $\frac{1}{2 \times f_{OS_{high}}}$).

Figure 4.4, shows the OTA circuit used in the DAC. Table 4.1 shows each transistor's size and bias current in the circuit.

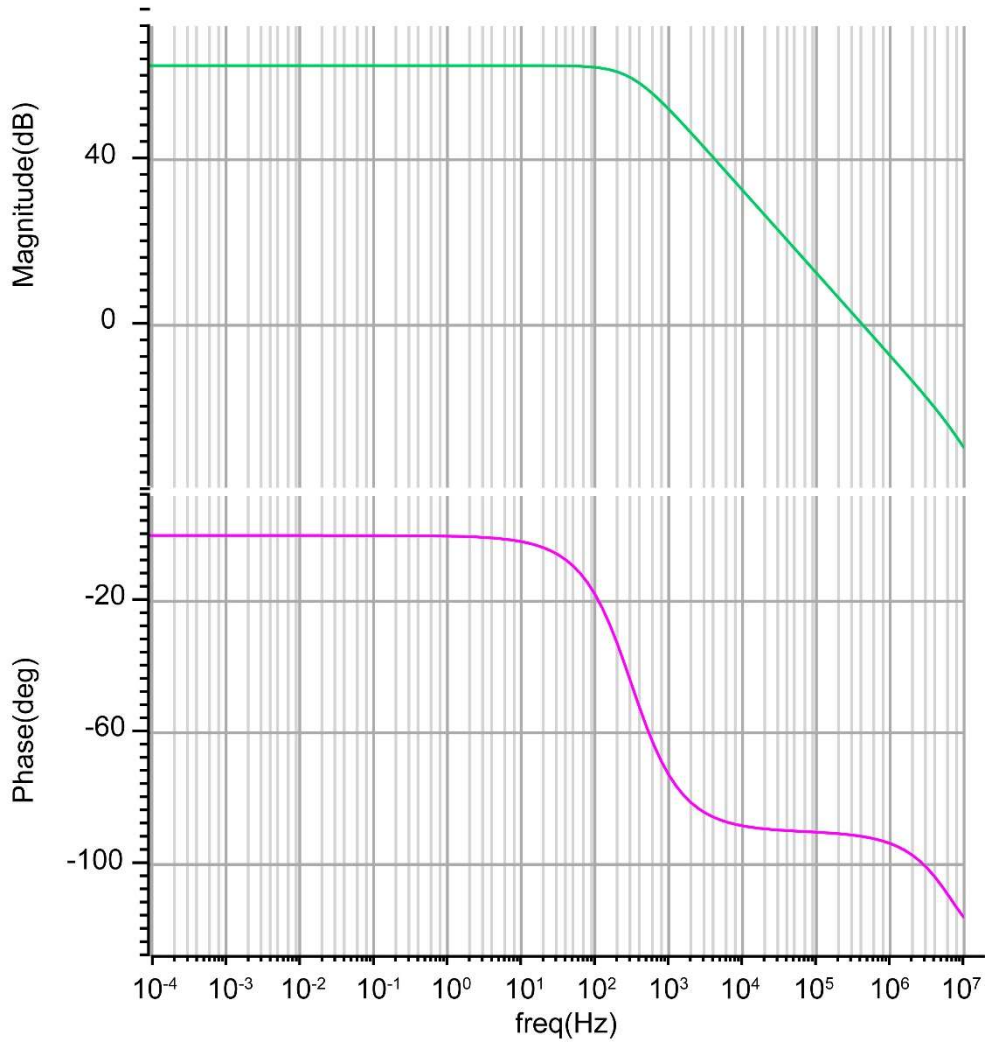


Figure 4.5 Frequency response of the designed OTA.

Table 4.2 Size and bias points of the DAC's OTA

MOSFET NO.	W/L	I_{BIAS} (A)
M1	1.13 μ / 1.1 μ	1.6 μ
M2	330n/230n	800n
M3	330n/230n	800n
M4	810n/1.13 μ	800n
M5	420n/2.3 μ	800n
M6	420n/2.3 μ	800n
M7	455n/9.13 μ	800n
M8	455n/9.13 μ	800n

Figure 4.5 depicts the frequency response of the OTA. DC gain is 62 dB, UGBW is 318kHz ($>2 \times f_{OS_{high}} = 128\text{kHz}$) and power consumption is $1.9\mu\text{W}$. At settling time, the OTA has a difference less than $5\mu\text{V}$ between its two input nodes, making the design compatible with the mentioned requirements.

4.2 Integrator circuit (G_m -C stage)

The input integrator in this design is a G_m -C stage, i.e., a voltage-to-current converting OTA followed by an integrating capacitor. The main requirement for the G_m -C stage is to ensure that it performs like an ideal integrator for the signal bandwidth of interest. This means that the 3-dB bandwidth (which is 0Hz in ideal integrators) should be small enough for system to be able to integrate and hold OSR number of samples (for $1/f_{NYQUIST}$ duration) to eliminate the accumulated error during this time.

In chapter 2, based on the SQNR equation, it was calculated that OSR=32 for input signal with $f_{IN_{MAX}} = 500\text{Hz}$ is sufficient to achieve 8-bit resolution.. However, in a non-ideal system, quantization noise is not the only noise added to the system (e.g., thermal and flicker noise of passive and active components), therefore to achieve the targeted resolution, a higher OSR is adopted to further reduce the quantization noise so that the targeted resolution can still be achieved after considering the other noise sources. The highest OSR in this design was chosen to be 64, thus the highest sampling rate of the system is equal to 64kHz ($64 \times 2 \times 500\text{Hz} = 64\text{kHz}$). Based on these values, the 3-dB bandwidth for the system is calculated as,

$$BW_{3dB} < \frac{f_{OS}}{OSR} = \frac{64\text{kHz}}{64} = 1\text{kHz} \quad (4.1)$$

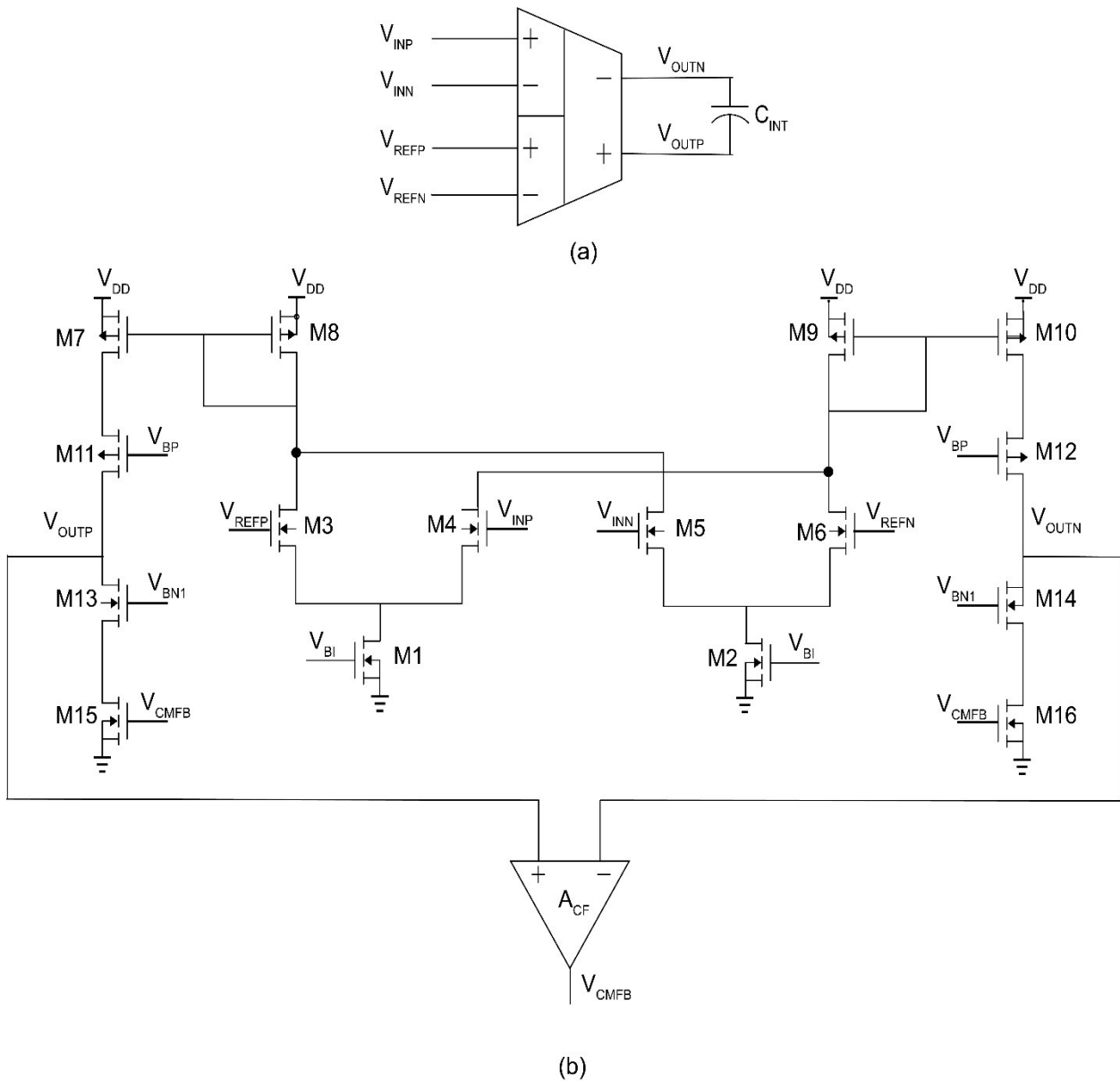


Figure 4.6 Detailed schematic of the G_m -C circuit

which is sufficient for recording bio-signals such as EEG, ECoG (Electrocorticogram) and iEEG signals with maximum frequency of a few hundred Hertz.

Figure 4.6 depicts the architecture of the G_m stage, which is implemented as a current mirror amplifier with a differential difference configuration. In this configuration the difference between the two input pairs $V_{INP} - V_{INN}$ and $V_{REFP} - V_{REFN}$ is amplified, hence it lowers the possibility of OTA being in its nonlinear region.

In order to choose the bias currents and sizes of the transistors, equations (4.2) to (4.4) which give an estimation for the total value of IRN power, output impedance, and BW_{3dB} of the OTA, respectively, should be taken into consideration.

$$V_{IN}^2 \approx 8KT\gamma \left(\frac{1}{g_{m3,4,5,6}} + \frac{g_{m1,2}}{g_{m3,4,5,6}^2} + \frac{2g_{m8,9}}{g_{m3,4,5,6}^2} + \frac{g_{m1,16}}{g_{m3,4,5,6}^2} \right) \quad (4.2)$$

$$R_{out} \approx \frac{r_{o1} r_{o15} g_{m13}}{2} \quad (4.3)$$

$$BW_{3dB} \approx \frac{1}{2\pi R_{out} (C_{out})} \quad (4.4)$$

In this design, the targeted integrated IRN for the G_m -C stage is aimed at $<5\mu V_{rms}$. This is to ensure that the total noise from the recording circuits is equal or less than the noise that already exists at the recording site (i.e., the background noise [37]), In addition to the input-referred noise, other design targets were a 3-dB bandwidth of 1kHz, and simultaneous maximization of input impedance and minimization of power consumption and area.

Using the g_m/id design approach, bias points were chosen in a way that: 1) input stage's transconductance efficiency is maximized, while the g_m of the other stages were reduced to minimize their effect on the input-referred thermal noise; 2) output stage's transistors' lengths (L) were increased to maximize the value of output impedance; and 3) input stage L were increased to reduce the effect of flicker noise, 4) input transistor were DC-coupled to the electrodes to both maximize the input impedance (i.e., gate impedance) and CMRR, and minimize the required area (no need for bulky decoupling capacitors [61]).

The DC-coupled inputs also allow for adding chopping switches at the input (for flicker noise and offset removal) without the concern for input impedance reduction due to input decoupling capacitors [62]. To complete the integrator, a capacitor (C_{INT} in Figure 4.6) was connected to the

differential output of the Gm stage. The capacitor value was chosen 500fF, resulting in a bandwidth of 20Hz. Table 4.2 shows the circuit's transistors bias currents and sizings.

Table 4.3 Size and bias points of G_m stage amplifier

MOSFET NO.	W/L	I_{BIAS} (A)
M1	1.9 μ /10 μ	500n
M2	1.9 μ /10 μ	500n
M3	3.3 μ /20 μ	250n
M4	3.3 μ /20 μ	250n
M5	3.3 μ /20 μ	250n
M6	3.3 μ /20 μ	250n
M7	11 μ /20 μ	500n
M8	11 μ /20 μ	500n
M9	11 μ /20 μ	500n
M10	11 μ /20 μ	500n
M11	11 μ /20 μ	500n
M12	11 μ /20 μ	500n
M13	2.6 μ /20 μ	500n
M14	2.6 μ /20 μ	500n
M15	557n /20 μ	500n
M16	557n /20 μ	500n

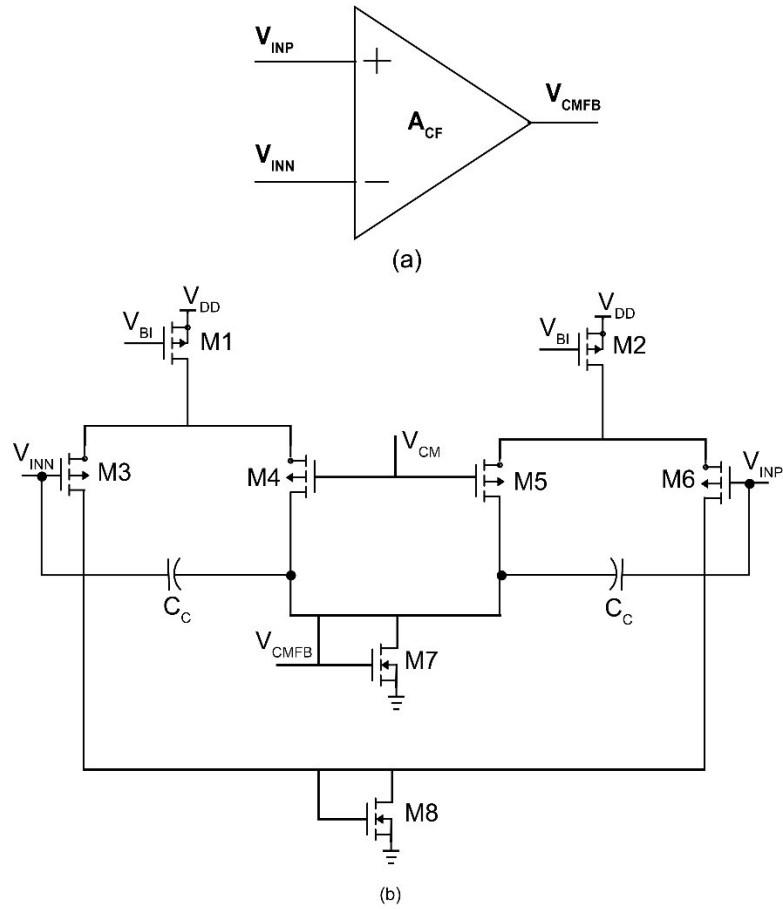


Figure 4.7 Detailed schematic of the (CMFB) circuit that is shown as; A_{CF} in the G_m - C circuit.

Table 4.4 Size and bias points of the CF amplifier

MOSFET NO.	W/L	$I_{BIAS}(A)$
M1	11.5 μ / 10 μ	500n
M2	11.5 μ / 10 μ	500n
M3	5.41 μ / 10 μ	250n
M4	5.41 μ / 10 μ	250n
M5	5.41 μ / 10 μ	250n
M6	5.41 μ / 10 μ	250n
M7	393n / 15 μ	500n
M8	393n / 15 μ	500n

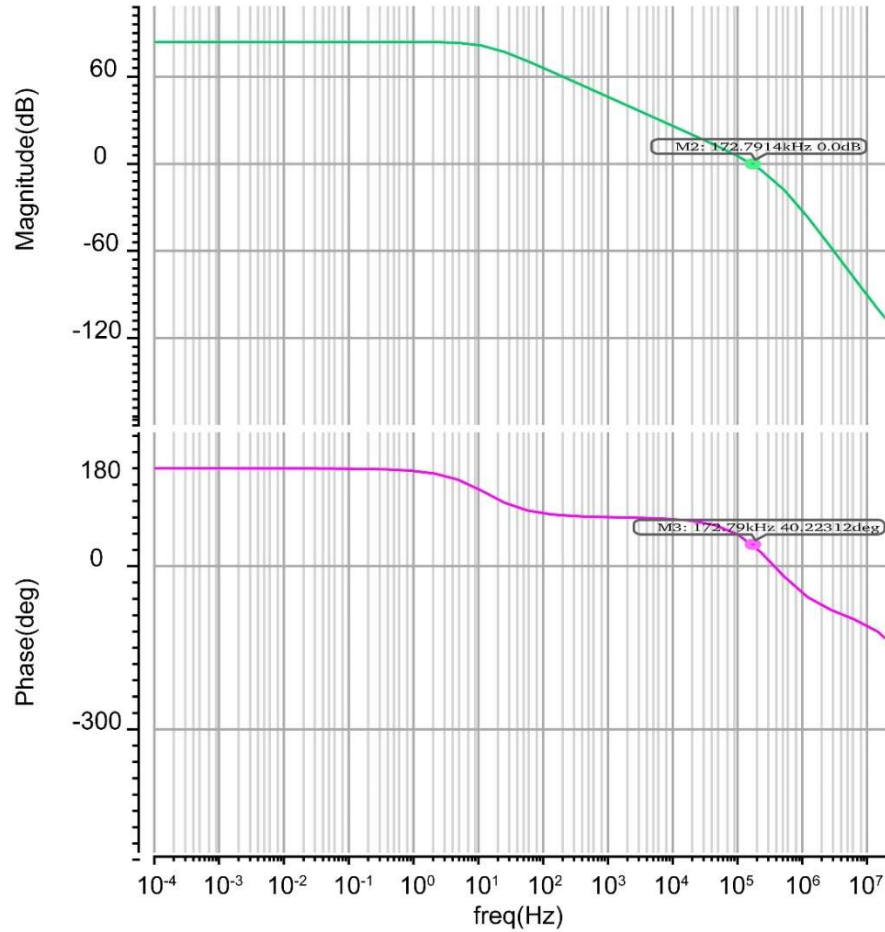


Figure 4.8 Frequency response of the designed G_m -C stage.

To ensure proper DC biasing at the differential outputs of the G_m stage, as shown in Figure 4.6, a common-mode feedback (CMFB) circuit was used. The CMFB circuit senses the DC at the output nodes and applies feedback to the gates of M15 and M16. Its detailed transistor-level schematic is presented in Figure 4.7. In addition, to stabilize the integrator, two 2pF compensation capacitors were used in the CMFB circuit. Table 4.3 shows CMFB circuit transistors bias currents and sizings.

Figure 4.8, shows the frequency response of the integrator. It has a DC gain of 83dB, with a 3-dB Bandwidth of 20Hz, and a phase margin of 40 degrees.

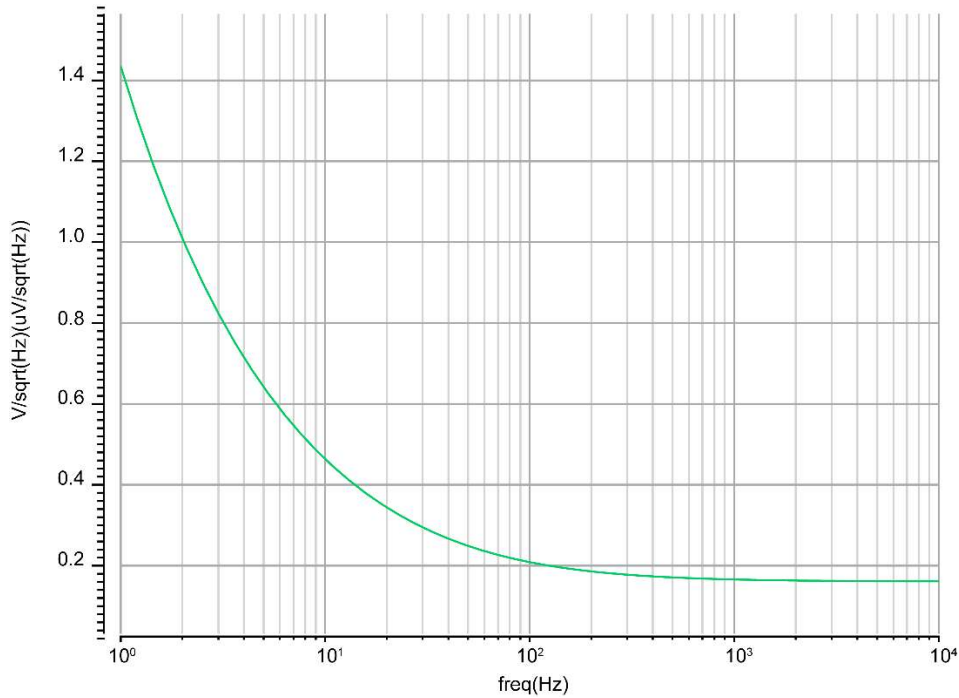


Figure 4.9 Power spectral density of the IRN of the designed $G_m - C$ stage.

Figure 4.9, depicts the power spectral density of the IRN of the $G_m - C$ stage. Corner frequency is around 32kHz, with noise floor of $161 \frac{nV}{\sqrt{Hz}}$. The integrated IRN over the band of interest (1-500Hz) is $4.8\mu V$, and the total power consumption is $3.6 \mu W$. Table 4.4 shows the performance parameters of the designed $G_m - C$ stage.

Table 4.5 performance parameter of the $G_m - C$ stage

Spec.

$V_{DD}(V)$	1.2
Power(W)	3.6μ
$R_{OUT} (\Omega)$	10G
$Z_{INCOM}(\Omega) (\sim 0 \text{ Hz})$	135G
$Z_{INDIFF}(\Omega) (\sim 0 \text{ Hz})$	216G
CMRR(dB)	>200
IRN($V_{rms} (1-500Hz)$)	4.8μ
Bandwidth (Hz)	500

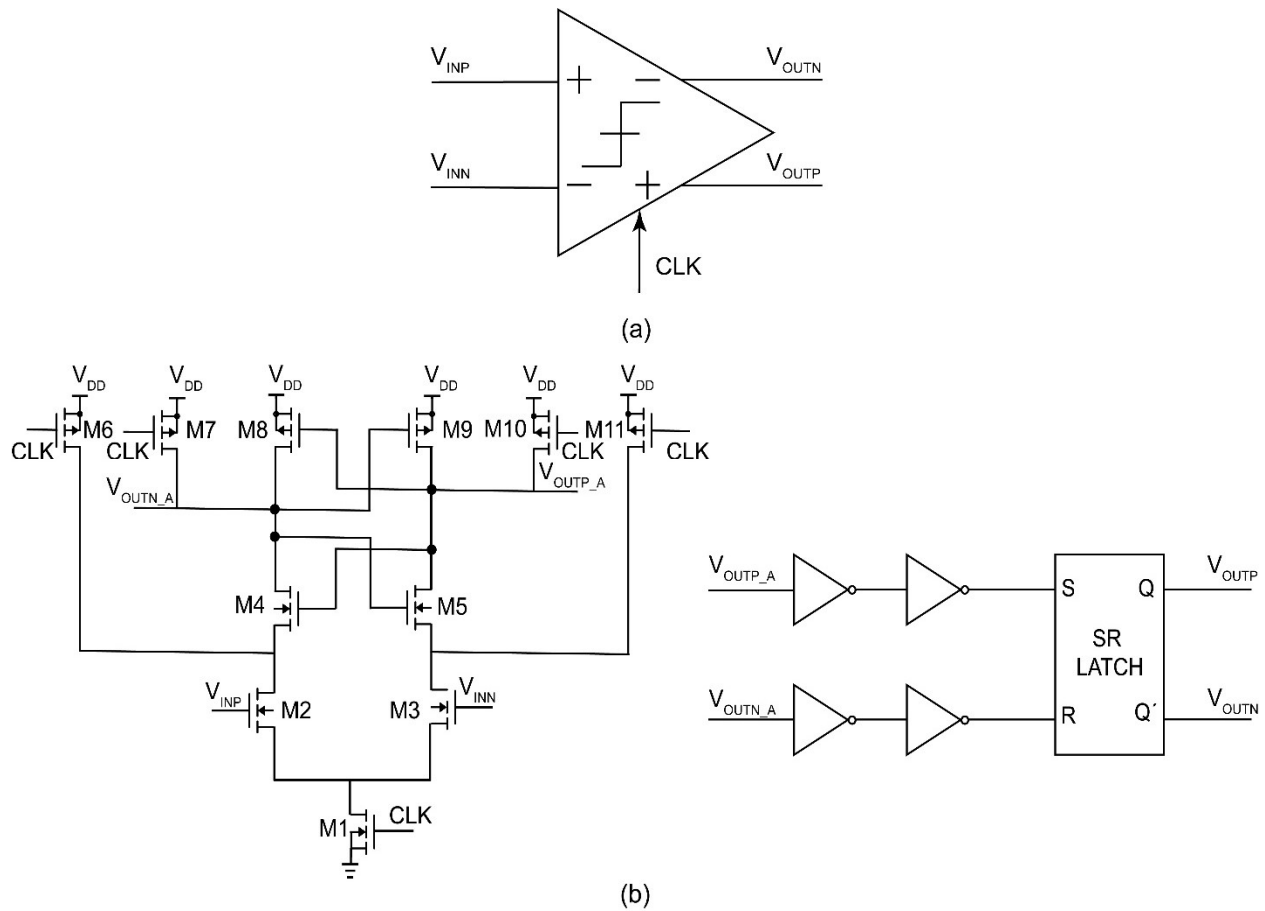


Figure 4.10 Transistor-level schematic of the StrongArm voltage comparator used in the proposed mixed-signal front-end.

4.3 Voltage comparator

StrongArm configuration was chosen to implement the 1-bit voltage comparator, shown in Figure 4.10. Our simulation results show a hysteresis band $< 7\mu\text{V}$, and conversion speed higher than 64kHz, which is higher than the modulator's highest targeted oversampling frequency. The average power consumption of the comparator is 16nW when clocked at max required speed of 64kHz.

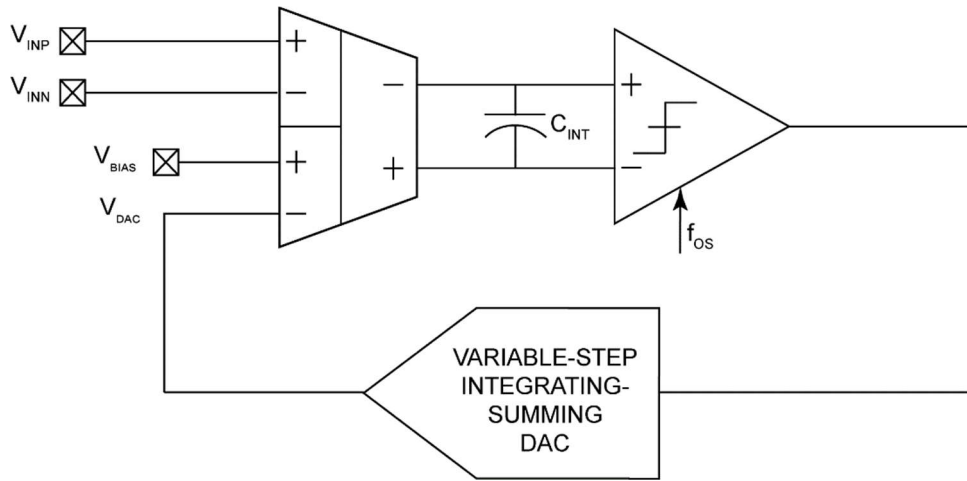


Figure 4.11 Block diagram of the $\Delta\Sigma$ modulator

4.4 Simulation results

Figure 4.11, depicts the overall block diagram of the presented modulator. Circuit-level simulation and verification were done using Cadence ViVA and MATLAB tool. In order to test the system, a sine wave (amplitude: 1mV, Frequency: 500Hz) with dynamics similar to a typical iEEG signal was fed to system as one of the inputs (V_{INP} , with $V_{BIAS} = V_{INN} = \frac{V_{DD}}{2}$), and the delayed estimated replica was observed to be successfully reconstructed by the modulator at the output of the summing-integrating feedback DAC (V_{DAC}).

Figure 4.12 shows the high-frequency output bitstream (i.e., the comparator's output; in this simulation due to the non-ideal effects introduced by the comparator in the 2nd order CT delta sigma modulator, which is the unwanted delayed introduced in the modulator with respect to DAC, the system-level simulation was conducted with an ideal comparator) as well as the input signal connected to the positive input and the reconstructed signal at the output of the feedback DAC.

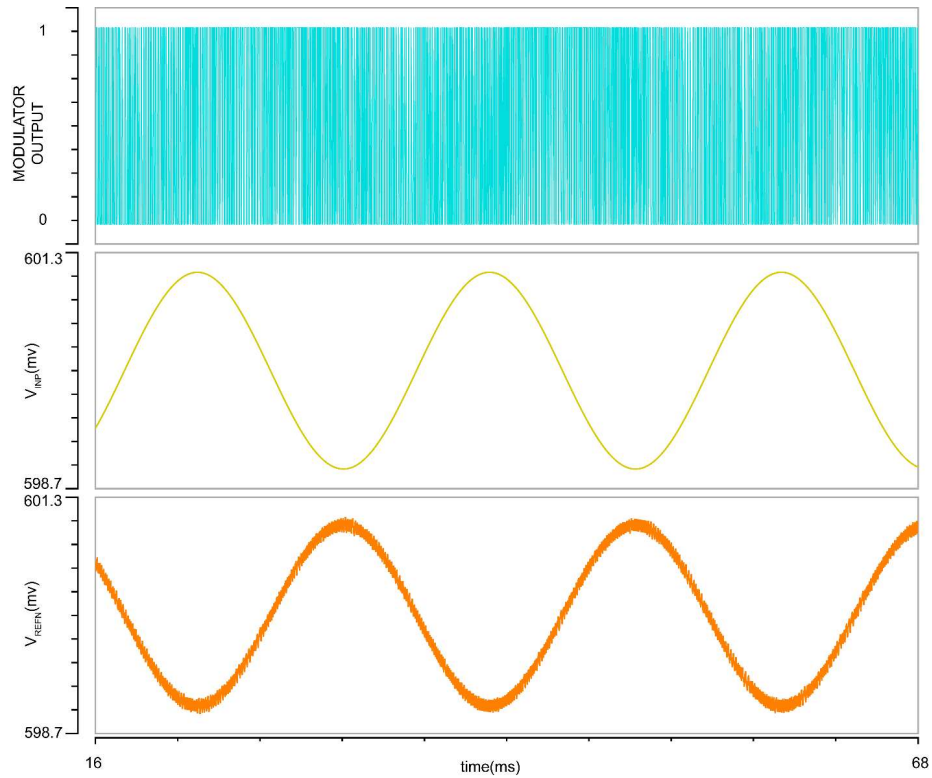


Figure 4.12 V_{REFN} delayed version of the input (V_{INP}) made from the Modulator's output using the proposed DAC

Figure 4.13 shows the zoomed version of these signals. It can be seen that how the step sizes are integrated to follow the input signal. As discussed earlier, it is expected if the bits are a sequence of [0,1] or [1,0], the step sizes should be 3 times higher than the reference step size, and when the bit sequence is [0,0] or [1,1] original reference step size should be applied. In Figure 4.13, it is shown that the step size is slightly higher than 3 times, (around 4 times). This is due to the fact that as mentioned earlier, in order to have the same result for DT and CT modulator, loop filter of both should have similar impulse response and that is done by adjusting “k” coefficients. Result of our simulations showed for a CT delta sigma, in order to have similar SNR results to DT counterpart, step sizes should increase around 4 times. This is equivalent of setting the correct coefficients for DAC functions, in order to have similar loop filters.

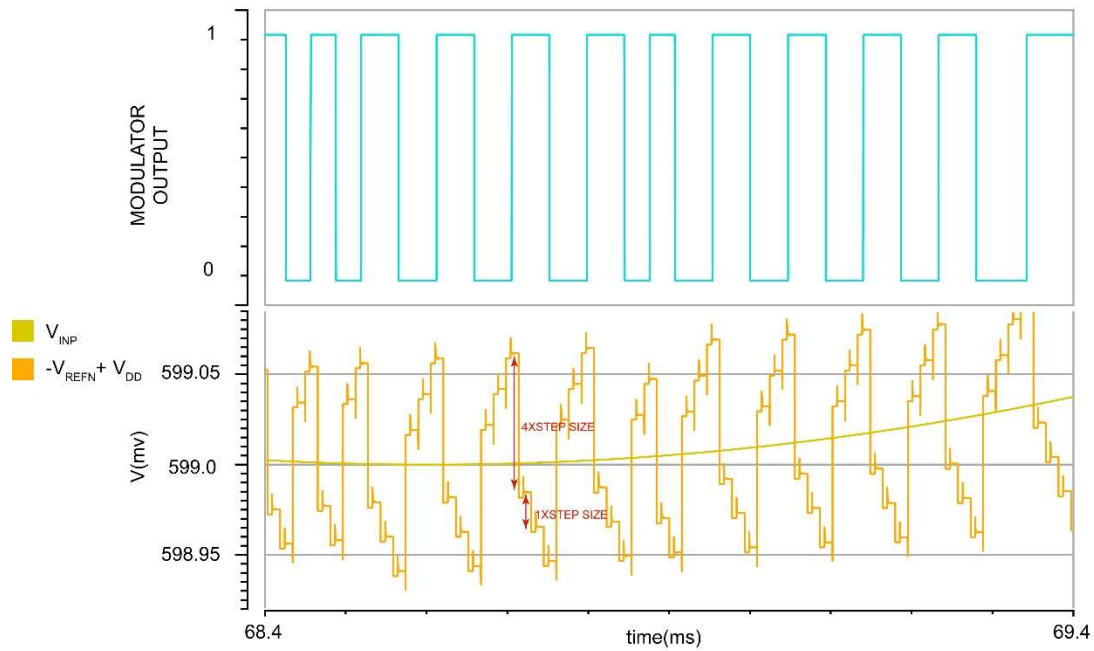


Figure 4.13 Step size integration according to modulators output bit stream sequence

Figure 4.14 depicts the system's input DC drift cancelation through the proposed DAC architecture. As it shown, differential input signal ($V_{INP}-V_{INN}$) with 50mV DC offset is closely followed by the differential feedback signal ($V_{BIAS}-V_{DAC}$) after around 40ms delay.

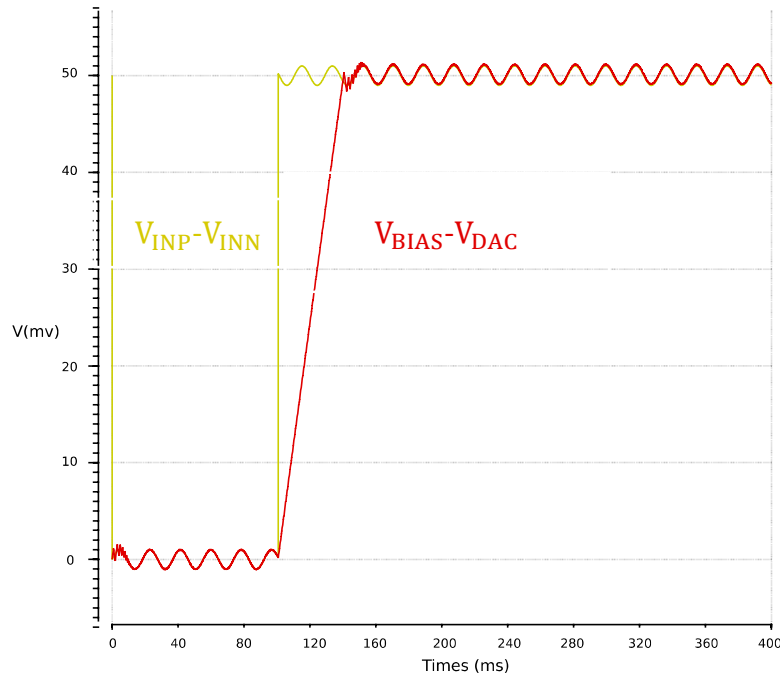


Figure 4.14 Input DC drift cancellation through the proposed DAC architecture

To verify the system's resolution, FFT of the modulator's output bitstream was taken for the two cases with target resolution of 4 bits and 8 bit. As mentioned earlier in this chapter, to take the effect of other noise sources into consideration, OSR was doubled; e.g., in theory for the target resolution of 8-bit resolution, OSR was calculated 32, however, for the nonideal system, OSR was increased to 64. Figures 4.14 and 4.15 show the FFT results for 4-bit and 8-bit resolution, respectively. The test was done by feeding a sine wave with frequency of $\sim 500\text{Hz}$ and V_{p-p} of 2mV (similar to the maximum of an iEEG signal's) to the modulator. For the 4-bit resolution, OSR=16 and $f_{OS}=16\text{kHz}$, and for the 8-bit resolution, OSR=64 and $f_{OS}=64\text{kHz}$ were selected. It can be seen in the FFT results that the SNR is obtained accordingly. As it was also previously mentioned, due to nonidealities of the feedback DAC, the integrated step sizes are not completely equal, therefore, the results include a small unwanted DC coefficient. The effect of this unwanted drift is eliminated by the digital back-end circuit using the baseline calculator unit (refer to chapter 3).

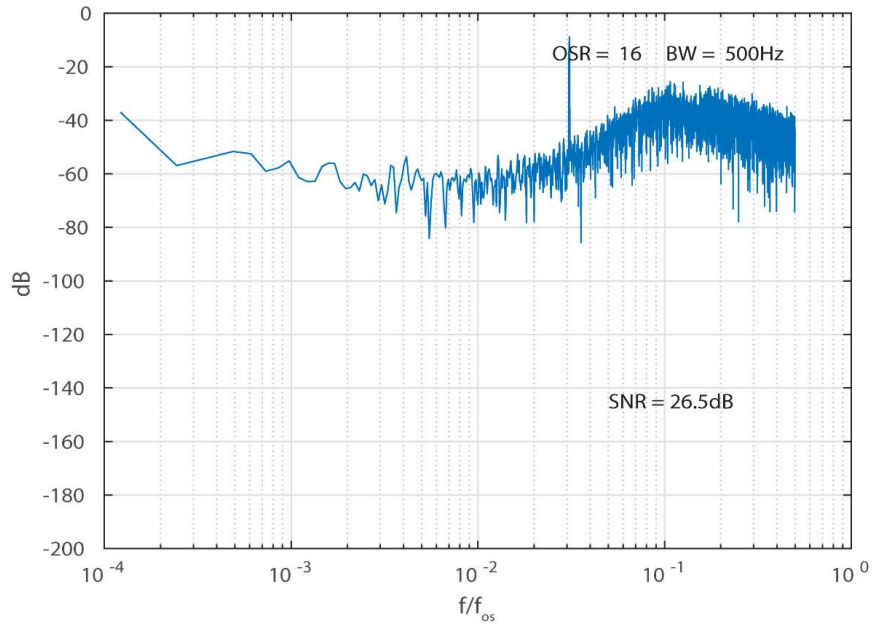


Figure 4.15 FFT of the modulator's output for OSR=16 (4-bit resolution)

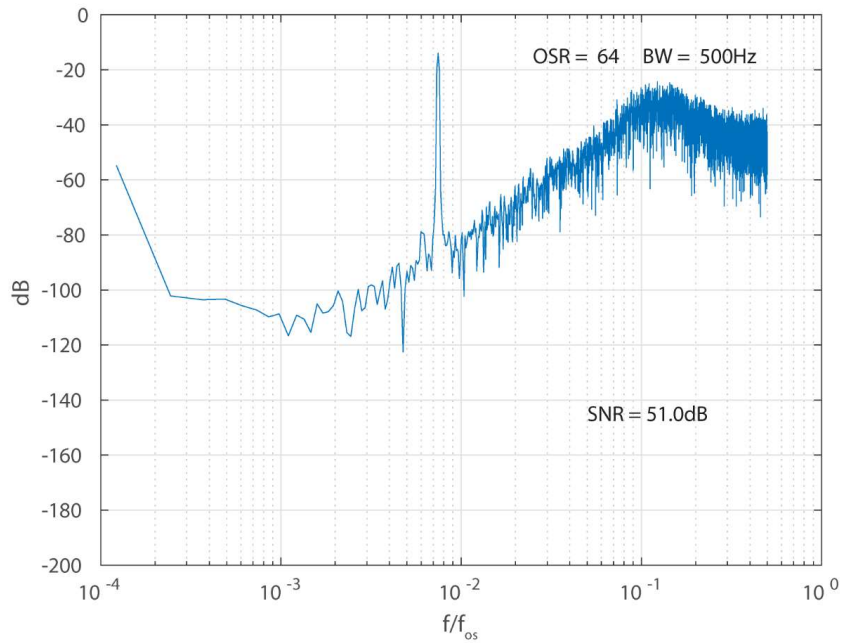


Figure 4.16 FFT of the modulator's output for OSR=64 (8-bit resolution)

Table 4.6 depicts the circuit-level result of the implemented circuit. The overall power consumption of the mixed-signal front-end is $5.6\mu\text{W}$ and overall input referred noise is $5.9(\mu\text{V}_{\text{RMS}})$

and maximum input dc drift is around $\pm 50\text{mV}$ which are close to the target performance in table 4.1. The result is also compared to several other similar works.

Table 4.6 Performance comparison with the state of the art.

Ref.	VLSI'17	JSSC'18	JSSC'17	This work
	[40]	[41]	[42]	
Process	180nm	65nm	40nm	130nm
V_{DD}(V)	1	0.8	1.2	1.2
Power/Channel(W)	8 μ	0.8	7 μ	5.6 μ
$Z_{IN_{DIFF}}$(Ω) (~ 0 Hz)	30M	NR	∞	216G
CMRR(dB)	NR	81	66	>200
Peak Input	100mV	>200mV	$\pm 50\text{mV}$	$\pm 50\text{mV}$
IRN (V_{rms})	1.6 μ	1.6 μ	5.2 μ	5.9 μ
Bandwidth (Hz)	DC-500	DC-500	1-200	1-500

* Results from this work are from Cadence simulations. Results from [40-42] are measurement results.

Chapter 5

Conclusions and Future Directions

5.1 Conclusion

This thesis presented the design, implementation, and characterization of a novel mixed-signal neural recording channel architecture for implantable BMIs that is capable of conducting input-activity-adaptive data compression, leading to significant energy efficiency improvement for implantable BMIs. Motivated by the growing demand for an ever-increasing number of neural recording channels, data compression for low power implantable sensing devices has been a popular research topic in recent years. We reviewed both lossless and lossy data compression techniques in this work and discussed their advantages and disadvantages in terms of compression factor, being generic vs application-specific, computation resource requirements, and data loss. It was shown how input-adaptive techniques offer a generic loss-less solution with a reasonable compression ratio and computational resource requirements. It was also shown how the benefits of such adaptive techniques could be fully leveraged when they are applied to fully-dynamic mixed-signal direct-ADC front-end architectures instead of conventional architectures.

Accordingly, a novel mixed-signal oversampling direct-ADC architecture with input-adaptive quantization resolution was proposed. First, the design and MATLAB-based functional implementation was discussed. It was shown theoretically how through adaptive resolution compression technique, the power consumption for transmission can be significantly reduced. Additionally, it was shown that by leveraging an oversampling Δ - $\Delta\Sigma$ within the architecture, it is possible to (a) remove the unwanted DC offsets and drifts at the input, which makes the front-end circuit needless of large DC coupling capacitors, take advantage of 2nd order noise shaping, which allows for significant reduction in the oversampling frequency, hence the overall dynamic power.

The MATLAB simulation results on a set of prerecorded human patients EEG recordings consisting of seizure episodes, showed that the proposed approach, at its simplest most-basic form of 2-level resolution, can achieve transmission power reduction by 43%, without losing any of the neurologically-relevant events. Significantly higher power reductions could be achieved through multiple-level resolution adjustment as well as more aggressive bit reduction, of course, at the cost of either a higher computational complexity, slight data loss, or higher power consumption.

Both the mixed-signal front-end and the digital backend modules of the proposed architecture were designed using Cadence and Synopsys CAD tools and a standard 130nm CMOS technology kit. The performance of the digital backend, responsible for input activity level evaluation and responsive control of the front-end module, was verified using post-layout simulations. It was shown that how system can detect the incoming signal's high activity intervals using a set of threshold bands, and also how it adjusts the front-end's oversampling clock frequency accordingly. The power consumption of the implemented back-end was measured to be 1.91 μ W and its total active area of 92 \times 92 μ m.

For the mixed-signal front-end, a novel architecture for the system's DAC was proposed in which one of the summing stages and integrator were merged and by adding appropriate voltage steps, the delayed version of the incoming signal was constructed with high precision. The results of simulation in VIVA and MATLAB showed that proposed front-end is capable of achieving both targeted quantization resolutions through simple clock frequency adjustment that is controlled by the activity evaluation unit in the digital backend module.

5.2 Statement on contributions

The proposed adaptive resolution algorithm development, its behavioral and system level implementation in MATLAB as well as HDL implementation are done by the thesis author.

The digital chip implementation from Verilog coding to physical layout and steps related to chip sign off are performed by the author.

The proposed novel neural ADC architecture as well as the proposed energy-efficient DAC built into it are contributions of the author.

All circuit designs, implementations, verifications and tests presented in this thesis are done by the author.

5.3 Future work

The proposed design shows a great potential for the low power implantable brain devices. However, the current state of the work has room for more improvement in order to be fully implemented and possibly commercialized in future. In the following sections some of the major improvements and also problems regarding the proposed design that can be investigated are discussed.

5.3.1 Improvement of the non-ideal effects on the front-end

As discussed in chapter 4, the proposed design is a CT 2nd order modulator. CT $\Delta\Sigma$ modulators, especially one with orders higher than one, introduce many non-idealities to the modulators performance. In this work, it was seen that due to the effect of delay between DAC and comparator, the performance of the modulator was degraded, and the results when using an ideal comparator in the circuit, were closer to the expected ones. A short-term objective is to trim the comparator stage performance in order to get the desired results for the system and physical implementation of the front-end.

In addition, due to adaptive nature of the system, the working clock frequency of system is continuously changing. When the rate of the frequency change significantly increases, there is a possibility of introduced nonlinearities in system's performance especially in the decimation filter in which the counting value will be changing with a high rate and so this could introduce unwanted effects in the reconstructed signal. One of the main focuses of the future works of this system is to further analyze and examine the system for extreme situation and the effect of nonlinearities seen in system due to that.

5.3.2 Improvement on the signal transient recovery

In the proposed design, it was shown that any kind of drift (unwanted DC on the neural signal) can be recovered with the proposed modulator. In general, it is desired to recover these drifts as fast as possible. In this current design, the recovery speed is heavily related to the size of the step that is chosen. The higher the step sizes, the faster the drift is canceled. In this design, it was shown that the reference step sizes are set based on a relation between the neural signal's amplitude, oversampling rate and nyquist frequency; in other words, the desired resolution is the determining

factor in choosing the step sizes. In order to overcome the drift, system can introduce a mechanism by which it significantly increases the step size temporary until the undesired drift is canceled. This mechanism need a predication unit that predicts an upcoming drift. This a unit that can notably improve the transient recovery speed of unwanted drifts on the signal.

5.3.3 Low power transmitter and data packaging

In this system, it was seen that the power is changed adaptively according to the signal's activity. This mechanism can be used to implement a custom integrated RF transmitters so that the transmitters power is also adaptively changed according to system's sampling rate. Proper data packaging techniques can be introduced to make data transmission with different resolution as efficient as possible.

5.3.4 Multi-channel implementation

Besides the above-mentioned features and many other features that could be added to the proposed architecture, a clear next step to show the efficacy of this design in a more realistic setting is to develop an IC that houses many (e.g., >100) of these channels as well as a wireless data transmitter to demonstrate its energy efficiency using in-vitro or in-vivo experiments.

Appendix A

A.1. MATLAB code of the described system

```
close all
```

```
%%%start of initialization
```

```
% read data
```

```
v_in1=data_array(486400:640000);
```

```
v_in = v_in1;
```

```
%up sampling data
```

```
up_sample_factor=125;
```

```
v_in = upsampler (v_in , up_sample_factor);
```

```
vin_size_vec = size(v_in);
```

```
vin_size = vin_size_vec (1,1);
```

```
v_amp = zeros (size(v_in));
```

```
v_out = zeros (size(v_in));
```

```
h_count= zeros (size(v_in));
```

```
%adc feedback step sizes
```

```
del_h=20;
```

```
del_l=80;

del=0;

%reconstruction in decimator

del_res_h=20;

del_res_l=20;

del_res=0;

integrator=0;

integrator_out=0;

s=vin_size;

out_bitstream=zeros(1, s);

outDAC=zeros(1, s);

FB_out=zeros(1, s);

FB_out_out=zeros(1, s);

reg=[0 0];

j=0;

down_sample_counter = 0 ;

up_sample_counter=0;10
```

```
DC_offset = 0 ;

DC_offset_record = zeros (size(v_in));

threshold_1 = 5;

threshold_2 = 10;

mid=0;

index=1;

time_step_size = 0;

Decimator_counter_integrator=zeros (size(v_in));

Decimator_counter_integrator(index)=0;

Quantizer_out = 0;

l=0;

h=0;

%%%End of initialization

%%%system (front-end and back-end)

while index<vin_size

    %ADC comparator output
```

```

comp_out=(v_in(index)- FB_out_out(index-time_step_size));

%Integrator in the forward path

integrator=integrator_out+comp_out;

%If the integrator in forward path exceeds 1.2v, it goes to saturation area

if(integrator>=1200000)

integrator_out=1200000;

elseif (integrator<=-1200000)

integrator_out=-1200000;

else

integrator_out=integrator;

end

%Quantization

if(integrator_out>=0)

    out_bitstream(index)=1;

    Quantizer_out = 1;

else

    out_bitstream(index)=0;

```

```

    Quantizer_out = -1 ;

end

%Two consecutive bits is stored

reg(2)=reg(1);

reg(1)=out_bitstream(index);

%Integrator output based on two consecutive bits

if( reg(2) ==0 && reg(1)==0)

    FB_out(index)=FB_out(index-time_step_size)-del;

elseif ( reg(2) ==0 && reg(1)==1)

    FB_out(index)=FB_out(index-time_step_size)+3*del;

elseif ( reg(2) ==1 && reg(1)==0)

    FB_out(index)=FB_out(index-time_step_size)-3*del;

elseif ( reg(2) ==1 && reg(1)==1)

    FB_out(index)=FB_out(index-time_step_size)+del;

end

%Feedback output saturation model

if(FB_out(index)>=1200000)

```

```

FB_out_out(index)=1200000;

elseif (FB_out(index)<=-1200000)

FB_out_out(index)=-1200000;

else

FB_out_out(index)=FB_out(index);

end

for i = 1:1:1000

FB_out_out (index+i) = FB_out_out (index) ;

end

for i = 1:1:1000

FB_out (index+i) = FB_out (index) ;

end

if(time_step_size==1)

Decimator_counter_integrator(index) = Decimator_counter_integrator (index-
time_step_size) + Quantizer_out ;

del_res=del_res_h;

end

if(time_step_size==4)

```

```
Decimator_counter_integrator(index) = Decimator_counter_integrator (index-  
time_step_size) + 4*Quantizer_out ;
```

```
del_res=del_res_l;
```

```
end
```

```
for i = 1:1:1000
```

```
Decimator_counter_integrator(index+i)= Decimator_counter_integrator(index) ;
```

```
end
```

```
if ( up_sample_counter>=31)
```

```
v_amp(index)=Decimator_counter_integrator(index);
```

```
v_out(index)=abs(del_res)*Decimator_counter_integrator(index);
```

```
for n = 1:1:10000
```

```
v_amp (index+n) = v_amp(index) ;
```

```
v_out(index+n)=v_out(index);
```

```
end
```

```
up_sample_counter=0;
```

```
down_sample_counter=down_sample_counter+1;
```

```
end
```

```
%Baseline calculator
```



```

if ( down_sample_counter==19)

    DC_offset = (7*DC_offset + Decimator_counter_integrator(index))/8;

    down_sample_counter=0;

    Decimator_counter_integrator(index)=Decimator_counter_integrator(index)- DC_offset ;

    for i = 1:1:125

        Decimator_counter_integrator(index+i)= Decimator_counter_integrator(index) ;

    end

    for n = 1:1:(10000*up_sample_factor)

        DC_offset_record (index+n) = DC_offset ;

    end

end

%Thershold levels

activity_threshold_high=threshold_2;

activity_threshold_midhigh=threshold_1;

activity_threshold_midlow=-1*threshold_1;

activity_threshold_low=-1*threshold_2;

%sampling freq. adjustment

```

```

if ((v_amp(index) > activity_threshold_high || v_amp(index) <= activity_threshold_low ) )

time_step_size = 1;

end

if ((v_amp(index) < activity_threshold_midhigh && v_amp(index) >
activity_threshold_midlow ) )

    time_step_size = 4;

end

if (( v_amp(index) > activity_threshold_midhigh && v_amp(index) <
activity_threshold_high) || ( v_amp(index) < activity_threshold_midlow && v_amp(index)
>= activity_threshold_low))

    mid=mid+1;

end

%stepsize adjustmnet

if(time_step_size==1)

    del=del_h;

    h=h+1;

    h_count(index)=1;

end

```

```

if(time_step_size==4)

    del=del_l;

    l=l+1;

    h_count(index)=0;

end

for i = 1:1:4

    h_count(index+i)= h_count(index);

end

end

%sampling

index = index + time_step_size ;

up_sample_counter=up_sample_counter+time_step_size;

end

%%%upsampler function used in initialization

function v_temp = upsampler (vin , over_sampling_factor )

[P,Q] = rat(over_sampling_factor);

v_temp = resample(vin,P,Q);

end

```

A.2. Verilog code for FPGA board to generate inputs for chip measurement.

```
module CHIP_TB ( input wire clk_in, clk_chip_in, reset_in,

                output reg ud_out, clk_out, reset_out,

                output reg [7:0]Th_l, Th_h,

                output reg [4:0] band,

                output reg [2:0] sel_low, sel_high, sel_nq,

                output wire led_test_l, led_test_h, led_test_b);

reg ram [0:1023];

reg [9:0] vector_counter_num;

reg [8:0] clk_counter_num, reset_counter;

%reading test vectors

initial

begin

$readmemb("C:/Users/KPS/Desktop/verilog_test/ICYKMS1_CHIP_TB/ICYKMS1_CHIP_TB/b

istream.txt", ram);

end

always @(posedge clk_in)
```

```

begin

    if( reset_in==0)

        begin

            clk_out<=1'b0;

            clk_counter_num<=9'b0;

            reset_counter<=9'b0;

            reset_out<=1'b0;

        end

    else

        begin

            if ( clk_counter_num == 9'b110000110 )

                begin

                    clk_out<=~clk_out;

                    clk_counter_num <=9'b0;

                end

            else

                begin

```

```

    clk_out<=clk_out;

    clk_counter_num<= clk_counter_num + 1'b1;

end

if ( reset_counter == 9'b111111110 && reset_out==1'b0 )

begin

reset_out<=1'b1;

end

else

begin

reset_counter<= reset_counter+ 1'b1;

end

end

end

always @ (posedge clk_out)

begin

    if ( reset_out == 0)

```

```

begin

    Th_l=8'b01110000;

    Th_h=8'b10010000;

    band=5'b01110;

    sel_low=3'b001;

    sel_high=3'b000;

    sel_nq=3'b101;

end

end

always @(posedge clk_chip_in)

begin

    if ( reset_out == 0)

begin

vector_counter_num <=10'b0;

ud_out <=0;

end

else

```

```

begin

if (vector_counter_num == 10'b111111111)

begin

ud_out<=0;

vector_counter_num <=10'b0;

end

else

begin

ud_out <= ram [ vector_counter_num ];

vector_counter_num <= vector_counter_num + 1'b1;

end

end

end

end

assign led_test_l = (Th_l == 8'b01110000 ) ? 1'b1 : 1'b0;

assign led_test_h = (Th_h == 8'b10010000 ) ? 1'b1 : 1'b0;

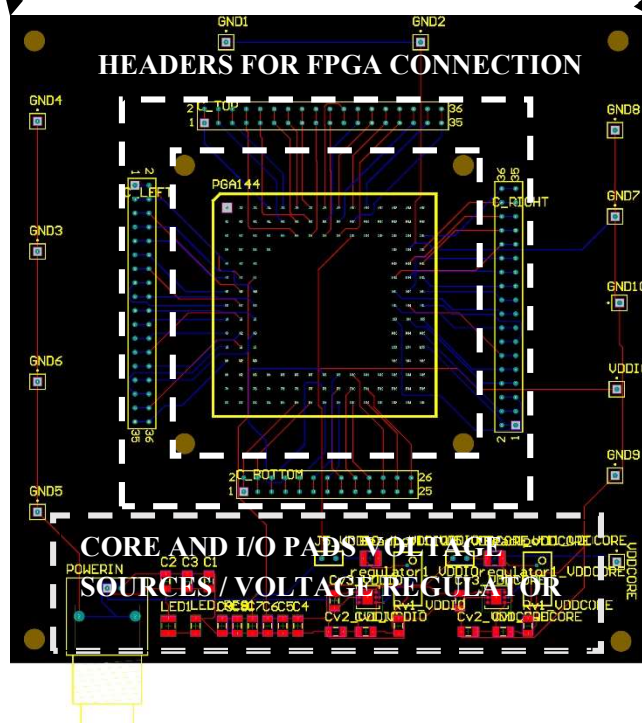
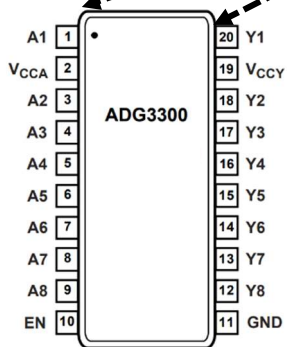
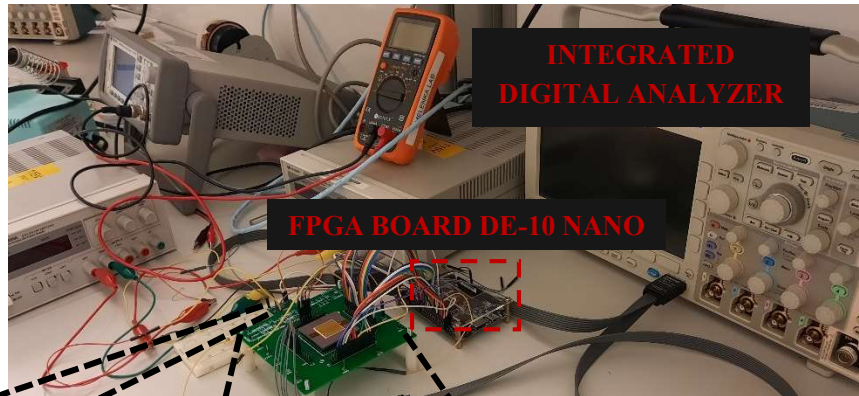
assign led_test_b = (band == 5'b01110 ) ? 1'b1 : 1'b0;

endmodule

```


A.3. Description of the measurement setup

The chip's core and pads are powered by 1.2V and 3.3V voltage respectively. An HDL test bench was written in Verilog (shown in A.2) and was implemented using FPGA de-10 nano board. Test vectors, i.e., threshold levels, hysteresis bands, bitstream, reset signal, and reference clock were generated using the FPGA and were fed to the chip. In addition, the bitstream test vector was generated from the result of EEG fed to the MATLAB. The bitstream test vector is saved in a memory array on board. Through a feedback mechanism, the chip will produce the oversampling frequency based on the inputs and feed it back to the FPGA board, and then memory array is read at the rate of oversampling frequency and is fed back to the chip. In addition, in order to be able to transfer data between FPGA and Chip, ADG3300 bidirectional level shifter to convert 1.2V and 3.3V voltage levels to each other, was used. All signals are measured using MD03022 200MHz Mixed Domain Oscilloscope with digital 16-bit bus probes. Figure A.3.1 shows the overall measurement set up.



A.3.1 Measurement set up annotation

Bibliography

- [1] World Health Organization. Neurological disorders: *public health challenges*. World Health Organization, 2006.
- [2] H. Kassiri et al., "Closed-Loop Neurostimulators: A Survey and a Seizure-Predicting Design Example for Intractable Epilepsy Treatment," *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 11, No. 5, pp. 1026-1040, Oct. 2017.
- [3] H. Kassiri et al., "Batteryless Tri-band-Radio Neuro-monitor and Responsive Neurostimulator for Diagnostics and Treatment of Neurological Disorders," *IEEE JSSC*, vol. 51, no. 5, pp. 1274-1289, May 2016.
- [4] M. R. Karimi et al., "A Multi-Feature Nonlinear-SVM Seizure Detection Algorithm with Patient-Specific Channel Selection and Feature Customization," *IEEE ISCAS*, pp. 1-5, 2020.
- [5] T. Zhan et al., "A resource-optimized VLSI implementation of a patient-specific seizure detection algorithm on a custom-made 2.2 cm² wireless device for ambulatory epilepsy diagnostics," *IEEE Transactions on Biomedical Circuits and Systems*, vol 13, no. 6, pp. 1175-1185, 2019.
- [6] F. Eshaghi, E. Najafiaghdam and H. Kassiri, "A 24-Channel Neurostimulator IC With Channel Specific Energy-Efficient Hybrid Preventive-Detective Dynamic-Precision Charge Balancing," in *IEEE Access*, vol. 9, pp. 95884-95895, 2021.
- [7] W. -M. Chen et al., "A Fully Integrated 8-Channel Closed-Loop Neural-Prosthetic CMOS SoC for Real-Time Epileptic Seizure Control," *IEEE JSSC*, vol. 49, no. 1, pp. 232-247, Jan. 2014.

- [8] M. A. Bin Altaf, C. Zhang and J. Yoo, "A 16-Channel Patient- Specific Seizure Onset and Termination Detection SoC With Impedance-Adaptive Transcranial Electrical Stimulator," *IEEE JSSC*, vol. 50, no. 11, pp. 2728-2740, Nov. 2015.
- [9] X. Liu et al., "A Fully Integrated Wireless Compressed Sensing Neural Signal Acquisition System for Chronic Recording and Brain Machine Interface," *IEEE TBioCAS*, vol. 10, no. 4, pp. 874-883, 2016.
- [10] S. Ha et al., "Silicon-Integrated High-Density Electro cortical Interfaces," in Proceedings of the IEEE, vol. 105, no. 1, pp. 11-33, Jan. 2017.
- [11] Safety Levels with Respect to Human Exposure to Radiofrequency Electromagnetic Fields", 3 kHz to 300 GHz, *ANSI/IEEE Standard C95.1-1992*
- [12] K. Sohee, P. Tathireddy, R.A. Normann, and F. Solzbacher, "Thermal impact of an active 3-d microelectrode array implanted in the brain," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 4, pp. 493–501, December 2007
- [13] H. Kassiri et al., "Rail-to-Rail-Input Dual-Radio 64- Channel Closed-Loop Neurostimulator," *IEEE JSSC*, vol. 52, no. 11, pp. 2793-2810, Nov. 2017.
- [14] X. Liu et al., "A Fully Integrated Wireless Compressed Sensing Neural Signal Acquisition System for Chronic Recording and Brain Machine Interface," *IEEE TBioCAS*, vol. 10, no. 4, pp. 874-883, 2016.
- [15] M. Shoaib, N. K. Jha and N. Verma, "A compressed-domain processor for seizure detection to simultaneously reduce computation and communication energy," Proceedings of the IEEE 2012 Custom Integrated Circuits Conference, 2012, pp. 1-4.

- [16] H. Bhamra et al., "An Ultra-Low Power 2.4 GHz Transmitter for Energy Harvested Wireless Sensor Nodes and Biomedical Devices," *IEEE TCAS-II: Express Briefs*, vol. 68, no. 1, pp. 206-210, Jan. 2021.
- [17] H. Bhamra et al., "A 24 μ W, Batteryless, Crystal-free, Multinode Synchronized SoC "Bionode" for Wireless Prosthesis Control," *IEEE JSSC*, vol. 50, no. 11, pp. 2714-2727, Nov. 2015.
- [18] P. P. Mercier et al., "A Sub-nW 2.4 GHz Transmitter for Low Data-Rate Sensing Applications," *IEEE JSSC*, vol. 49, no. 7, pp. 1463-1474, 2014.
- [19] G. Papotto et al., "A 90-nm CMOS 5-Mbps Crystal-Less RF Powered Transceiver for Wireless Sensor Network Nodes," *IEEE JSSC*, vol. 49, no. 2, pp. 335-346, Feb. 2014.
- [20] Chien-Hua Jung et al., "A 0.9-V 2.36-GHz MedRadio-band 10-Mbps low-power OOK modulator for neural implants," *2017 International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*, pp. 1-4, 2017,
- [21] Z. Wang et al., "A Software-Defined Always-On System With 57–75-nW Wake-Up Function Using Asynchronous Clock-Free Pipelined Event-Driven Architecture and Time-Shielding Level-Crossing ADC," in *IEEE Journal of Solid-State Circuits*, vol. 56, no. 9, pp. 2804-2816, Sept. 2021.
- [22] G. Higgins et al., "The Effects of Lossy Compression on Diagnostically Relevant Seizure Information in EEG Signals," in *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 1, pp. 121-127, Jan. 2013.
- [23] M. Shoaib, "Design of Energy-efficient Sensing Systems with Direct Computations on Compressively-sensed Data," Ph.D. dissertation, Dept. of Elec. Eng. , PU, Princeton, NJ, USA, 2013.

- [24] M. Wakin et al., "An architecture for compressive imaging," Proc. IEEE Int. Conf. Image Processing, pp. 1273–1276, Oct.2006.
- [25] Z. Chaozhu and L. Jing, "Distributed video coding based on compressive sensing," *Int. Conf. Multimedia Technology*, pp.3046–3049, Jul. 2011.
- [26] D. Gangopadhyay, E. G. Allstot, A. M. R. Dixon, K. Natarajan, S. Gupta and D. J. Allstot, "Compressed Sensing Analog Front-End for Bio-Sensor Applications," in *IEEE Journal of Solid-State Circuits*, vol. 49, no. 2, pp. 426-438, Feb. 2014.
- [27] F. Chen, A. P. Chandrakasan and V. M. Stojanovic, "Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors," in *IEEE Journal of Solid-State Circuits*, vol. 47, no. 3, pp. 744-756, March 2012.
- [28] S. Kirolos et al., "Analog-to-Information Conversion via Random Demodulation," *2006 IEEE Dallas/CAS Workshop on Design, Applications, Integration and Software*, 2006, pp. 71-74.
- [29] F. Chen, A. P. Chandrakasan and V. Stojanović, "A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors," *IEEE Custom Integrated Circuits Conference 2010*, 2010, pp. 1-4.
- [30] W. Zhao, B. Sun, T. Wu and Z. Yang, "On-Chip Neural Data Compression Based On Compressed Sensing With Sparse Sensing Matrices," in *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 1, pp. 242-254, Feb. 2018.
- [31] F. Pareschi, P. Albertini, G. Frattini, M. Mangia, R. Rovatti and G. Setti, "Hardware-Algorithms Co-Design and Implementation of an Analog-to-Information Converter for Biosignals

Based on Compressed Sensing," in *IEEE Transactions on Biomedical Circuits and Systems*, vol. 10, no. 1, pp. 149-162, Feb. 2016.

[32] H. Kassiri et al., " All-wireless 64-channel 0.013 mm² /ch closed-loop neurostimulator with rail-to-rail DC offset removal ", *Proc. IEEE Int. Solid-State Circuits Conf.*, pp. 452-453, Feb. 2017.

[33] T. Moeinfard, G. Zoidl and H. Kassiri, "A SAR-Assisted DC-Coupled Chopper-Stabilized 20 μ s-Artifact-Recovery $\Delta\Sigma$ ADC for Simultaneous Neural Recording and Stimulation," *2022 IEEE Custom Integrated Circuits Conference (CICC)*, 2022, pp. 1-2.

[34] Shanthi Pavan; Richard Schreier; Gabor C. Temes, *Understanding Delta-Sigma Data Converters*, 2nd Edition, Wiley professional, Reference & Trade, pp. 223-238, 2017

[35] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation* [Online]. 101 (23), pp. e215–e220.

[36] Brundvand, E., *Digital VLSI Chip Design with Cadence and Synopsys CAD Tools*, Addison-Welsey, 2010

[37] Moo Sung Chae, Wentai Liu, and M. Sivaprakasam, "Design Optimization for Integrated Neural Recording Systems," *Solid-State Circuits, IEEE Journal of*, vol. 43, no. 9, pp. 1931–1939, Sept. 2008.

[38] M. T. Salam, H. Kassiri, Nima Soltani, Haoyu He, Jose Luis Perez Velazquez, and Roman Genov. "Tradeoffs between wireless communication and computation in closed-loop implantable devices." In *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1838-1841. IEEE, 2016.

- [39] Rezaee Kh., Azizi E., Haddadnia J. "Optimized Seizure Detection Algorithm: A Fast Approach for Onset of Epileptic in EEG Signals Using GT Discriminant Analysis and K-NN Classifier." *Journal of Biomedical Physics & Engineering*. 2016, pp. 81-94.
- [40] B. C. Johnson et al., "An implantable 700 μ W 64-channel neuromodulation IC for simultaneous recording and stimulation with rapid artifact recovery," *2017 Symposium on VLSI Circuits, 2017*, pp. C48-C49.
- [41] C. Kim, S. Joshi, H. Courellis, J. Wang, C. Miller and G. Cauwenberghs, "Sub-Vrms-Noise Sub- μ W/Channel ADC-Direct Neural Recording With 200-mV/ms Transient Recovery Through Predictive Digital Autoranging," in *IEEE Journal of Solid-State Circuits*, vol. 53, no. 11, pp. 3101-3110, Nov. 2018.
- [42] W. Jiang, V. Hokyikyan, H. Chandrakumar, V. Karkare and D. Marković, "A \pm 50-mV Linear-Input-Range VCO-Based Neural-Recording Front-End With Digital Nonlinearity Correction," in *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, pp. 173-184, Jan. 2017.
- [43] Ha, Sohmyung, Abraham Akinin, Jiwoong Park, Chul Kim, Hui Wang, Christoph Maier, Patrick P. Mercier, and Gert Cauwenberghs. "Silicon-integrated high-density electrocortical interfaces." *Proceedings of the IEEE* 105, no. 1 (2016): 11-33.
- [44] M. Judy, M. S. Amir and R. Lotfi, "A nonlinear signal-specific ADC for efficient neural recording," *2010 Biomedical Circuits and Systems Conference (BioCAS)*, 2010, pp. 17-20.
- [45] J. Van Rethy, M. De Smedt, M. Verhelst and G. Gielen, "Predictive sensing in analog-to-digital converters for biomedical applications," *International Symposium on Signals, Circuits and Systems ISSCS2013*, 2013, pp. 1-4.

- [46] M. Trakimas and S. R. Sonkusale, "An Adaptive Resolution Asynchronous ADC Architecture for Data Compression in Energy Constrained Sensing Applications," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 5, pp. 921-934, May 2011.
- [47] Chemparathy, Aditi, et al. "Wearable low-latency sleep stage classifier." *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*. IEEE, 2014.
- [48] Li, Peter Zhi Xuan, Hossein Kassiri, and Roman Genov. "A compact low-power VLSI architecture for real-time sleep stage classification." *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2016.
- [49] Yousefi, Tayebbeh, et al. "An Energy-Efficient Optically-Enhanced Highly-Linear Implantable Wirelessly-Powered Bidirectional Optogenetic Neuro-Stimulator." *IEEE Transactions on Biomedical Circuits and Systems* 14.6 (2020): 1274-1286.
- [50] Kassiri, Hossein, et al. "Inductively powered arbitrary-waveform adaptive-supply electro-optical neurostimulator." *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2015.
- [51] Salam, Muhammad T., et al. "Rapid brief feedback intracerebral stimulation based on real-time desynchronization detection preceding seizures stops the generation of convulsive paroxysms." *Epilepsia* 56.8 (2015): 1227-1238.
- [52] Dabbaghian, Alireza, et al. "A 9.2-g fully-flexible wireless ambulatory EEG monitoring and diagnostics headband with analog motion artifact detection and compensation." *IEEE transactions on biomedical circuits and systems* 13.6 (2019): 1141-1151.

- [53] Dabbaghian, Alireza, et al, "An 8-Channel 0.45 mm²/Channel EEG Recording IC with ADC-Free Mixed-Signal In-Channel Motion Artifact Detection and Removal." *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2020.
- [54] Kassiri, Hossein, et al. "Battery-less modular responsive neurostimulator for prediction and abortion of epileptic seizures." *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2016.
- [55] Taghadosi, Mansour, and Hossein Kassiri. "A Calibration-Free Energy-Efficient IC for Link-Adaptive Real-Time Energy Storage Optimization of CM Inductive Power Receivers." *IEEE Journal of Solid-State Circuits* 57.3 (2021): 793-802.
- [56] Soltani, Nima, et al. "0.13 μm CMOS 230Mbps 21pJ/b UWB-IR transmitter with 21.3% efficiency." *ESSCIRC Conference 2015-41st European Solid-State Circuits Conference (ESSCIRC)*. IEEE, 2015.
- [57] Sayedi, Mina, and Hossein Kassiri. "Activity-Adaptive Architectures for Energy-Efficient Scalable Neural Recording Microsystems: A Review of Current and Future Directions." *2022 20th IEEE Interregional NEWCAS Conference (NEWCAS)*. IEEE, 2022.
- [58] Pazhouhandeh, M. Reza, et al. "Artifact-tolerant opamp-less delta-modulated bidirectional neuro-interface." *2018 IEEE symposium on VLSI Circuits*. IEEE, 2018.
- [59] Bialer, Meir, et al. "Seizure detection and neuromodulation: A summary of data presented at the XIII conference on new antiepileptic drug and devices (EILAT XIII)." *Epilepsy research* 130 (2017): 27-36.
- [60] Zhan, Tianyu, Sam Guraya, and Hossein Kassiri. "A resource-optimized VLSI architecture for patient-specific seizure detection using frontal-lobe EEG." *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2019.

[61] Kassiri, Hossein, Karim Abdelhalim, and Roman Genov. "Low-distortion super-GOhm subthreshold-MOS resistors for CMOS neural amplifiers." *2013 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2013.

[62] Kassiri, Hossein, et al. "Inductively-powered direct-coupled 64-channel chopper-stabilized epilepsy-responsive neurostimulator with digital offset cancellation and tri-band radio." *ESSCIRC 2014-40th European Solid State Circuits Conference (ESSCIRC)*. IEEE, 2014.