Hugo Alexandre de Andrade Serra

Mestre em Engenharia Electrotécnica e de Computadores

Analysis and Design Methodologies for Switched-Capacitor Filter Circuits in Advanced CMOS Technologies

Dissertação para obtenção do Grau de
Doutor em Engenharia Electrotécnica e de Computadores

Orientador:	Nuno Filipe Silva Veríssimo Paulino, Prof. Auxiliar, Universidade Nova de Lisboa
Co-orientador:	Rui Manuel Leitão Santos Tavares, Prof. Auxiliar, Universidade Nova de Lisboa

Júri:	
Presidente:	Paulo da Costa Luís da Fonseca Pinto
Arguentes:	Nuno Cavaco Gomes Horta
	João Pedro Abreu de Oliveira
Vogais:	Jorge Manuel Correia Guilherme
	Andrzej Jaszkiewicz
	Nuno Filipe Silva Veríssimo Paulino

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Dezembro, 2017

Analysis and Design Methodologies for Switched-Capacitor Filter Circuits in Advanced CMOS Technologies

Acknowledgements

First of all, I would like to thank my supervisor, Prof. Nuno Paulino, for his dedication, encouragement, and patience throughout the five years of my Ph.D., without his help the work presented in this thesis would not have been possible. I am also grateful to my co-supervisor, Prof. Rui Tavares, for his help and guidance in the development of the optimization software.

I would also like to thank Prof. Manuel Medeiros Silva, for his constructive criticism regarding the structure, organization and writing of my scientific papers; the professors of the electronics group, Prof. João Goes, Prof. João Pedro Oliveira, Prof. Luís Oliveira, and Prof. Maria Helena Fino, for their availability and support; and Nuno Correia for his technical support with the simulation server.

I gratefully acknowledge the financial support received towards my Ph.D. from the Portuguese Foundation for Science and Technology under a Ph.D. Grant (SFRH/BD/87476/2012) and from the European Union's H2020 Programme through project PROTEUS under grant agreement No. 644852. I also acknowledge the Electrical Engineering Department of FCT/UNL and CTS-UNINOVA for providing the necessary conditions to perform the research work.

I am grateful to my colleagues and friends, Edinei Santin, Ivan Bastos, João de Melo, Błażej Nowacki, Somayeh Abdollahvand, Diogo Inácio, Nuno Pereira, Ana Correia, Miguel Fernandes, Ricardo Madeira, Miguel Teixeira and Rogério Rebelo for all the support.

Last but not least I wish to thank my family for their continuous support over the years.

ABSTRACT

Analog filters are an extremely important block in several electronic systems, such as RF transceivers, data acquisition channels, or sigma-delta modulators. They allow the suppression of unwanted frequencies bands in a signal, improving the system's performance. These blocks are typically implemented using active RC filters, g_m-C filters, or switched-capacitor (SC) filters.

In modern deep-submicron CMOS technologies, the transistors intrinsic gain is small and has a large variability, making the design of moderate and high-gain amplifiers, used in the implementation of filter blocks, extremely difficult. To avoid this difficulty, in the case of SC filters, the opamp can be replaced with a voltage buffer or a low-gain amplifier (< 2), simplifying the amplifier's design and making it easier to achieve higher bandwidths, for the same power. However, due to the loss of the virtual ground node, the circuit becomes sensitive to the effects of parasitic capacitances, which effect needs to be compensated during the design process.

This thesis addresses the task of optimizing SC filters (mainly focused on implementations using low-gain amplifiers), helping designers with the complex task of designing high performance SC filters in advanced CMOS technologies. An efficient optimization methodology is introduced, based on hybrid cost functions (equation-based/simulation-based) and using genetic algorithms.

The optimization software starts by using equations in the cost function to estimate the filter's frequency response reducing computation time, when compared with the electrical simulation of the circuit's impulse response. Using equations, the frequency response can be quickly computed (< 1 s), allowing the use of larger populations in the genetic algorithm (GA) to cover the entire design space. Once the specifications are met, the population size is reduced and the equation-based design is fine-tuned using the more computationally intensive, but more accurate, simulation-based cost function, allowing to accurately compensate the parasitic capacitances, which are harder to estimate using equations. With this hybrid approach, it is possible to obtain the final optimized design within a reasonable amount of computation time.

Two methods are described for the estimation of the filter's frequency response. The first method is hierarchical in nature where, in the first step, the frequency response is optimized using the circuit's ideal transfer function. The following steps are used to optimize circuits, at transistor level, to replace the ideal blocks (amplifier and switches) used in the first step, while compensating the effects of the circuit's parasitic capacitances in the ideal design. The second method uses a novel efficient numerical methodology to obtain the frequency response of SC filters, based on the circuit's first-order differential equations. The methodology uses a non-hierarchical approach, where the non-ideal effects of the

transistors (in the amplifier and in the switches) are taken into consideration, allowing the accurate computation of the frequency response, even in the case of incomplete settling in the SC branches.

Several design and optimization examples are given to demonstrate the performance of the proposed methods. The prototypes of a second order programmable bandpass SC filter and a 50 Hz notch SC filter have been designed in UMC 130 nm CMOS technology and optimized using the proposed optimization software with a supply voltage of 0.9 V. The bandpass SC filter has a total power consumption of 249 μW. The filter's central frequency can be tuned between 3.9 kHz and 7.1 kHz, the gain between -6.4 dB and 12.6 dB, and the quality factor between 0.9 and 6.9. Depending on the bit configuration, the circuit's THD is between -54.7 dB and -61.7 dB. The 50 Hz notch SC filter has a total power consumption of 273 μW. The transient simulation of the circuit's extracted view (C+CC) shows an attenuation of 52.3 dB in the 50 Hz interference and that the desired 5 kHz signal has a THD of -92.3 dB.

Keywords: Bandpass filter, computer-aided design, Fleischer-Laker topology, genetic algorithms, hybrid cost function, low-gain amplifiers, lowpass filter, notch filter, Sallen-Key topology, switched-capacitor filters.

RESUMO

Filtros analógicos são um bloco extremamente importante em vários sistemas eletrónicos, como transceptores de rádio frequência, canais de aquisição de dados ou modeladores sigma-delta. Eles permitem a supressão de bandas de frequências indesejadas num sinal, melhorando o desempenho do sistema. Estes blocos são geralmente implementados utilizando filtros RC activos, filtros g_m-C ou filtros de condensadores comutados.

Nas tecnologias CMOS submicrométricas modernas, o ganho intrínseco dos transístores é pequeno e tem uma grande variabilidade, tornando extremamente difícil o projecto de amplificadores de ganho moderado e elevado, que são necessários para a implementação de blocos de filtragem. Para evitar esta dificuldade, no caso de filtros de condensadores comutados, o amplificador pode ser substituído por um seguidor de tensão ou por um amplificador de ganho reduzido (< 2), simplificando o projecto do amplificador e tornando mais fácil atingir larguras de banda mais elevadas para a mesma potência. No entanto, devido à perda do nó de massa virtual, o circuito torna-se sensível aos efeitos das capacidades parasitas, cujo efeito é necessário compensar durante a fase de dimensionamento.

Esta tese aborda a tarefa de optimizar filtros de condensadores comutados (focando-se principalmente em implementações com amplificadores de ganho reduzido), ajudando engenheiros na tarefa complexa de projectar filtros de condensadores comutados com alto desempenho em tecnologias CMOS avançadas. Uma metodologia de optimização eficiente é introduzida, com base em funções de custo híbridas (baseada em equações/simulações) e utilizando algoritmos genéticos.

O software de optimização começa por utilizar equações na função de custo para estimar a resposta em frequência do filtro, reduzindo o tempo de computação quando comparado com a simulação eléctrica da resposta ao impulso do circuito. Utilizando equações, a resposta em frequência pode ser calculada rapidamente (< 1 s), permitindo o uso de populações maiores no algoritmo genético para cobrir todo o espaço de busca. Uma vez que as especificações sejam cumpridas, o tamanho da população é reduzido e o dimensionamento obtido através de equações é ajustado utilizando simulações transientes na função de custo. Embora seja computacionalmente intensivo, as simulações permitem compensar com precisão as capacidades parasitas, que são mais difíceis de estimar através de equações. Com esta abordagem híbrida, é possível obter o dimensionamento final optimizado num tempo de computação razoável.

Dois métodos são descritos para estimar a resposta em frequência do filtro. O primeiro método usa uma abordagem hierárquica onde, inicialmente, a resposta em frequência é optimizada recorrendo à função de transferência ideal do circuito. As etapas seguintes são utilizadas para optimizar os circuitos,

ao nível do transístor, para substituir os blocos ideais (amplificador e interruptores) utilizados no primeiro passo, compensando os efeitos das capacidades parasitas destes blocos no dimensionamento ideal. O segundo método utiliza uma nova metodologia numérica e eficiente para obter a resposta em frequência de filtros de condensadores comutados, com base nas equações diferenciais de primeira ordem do circuito. A metodologia usa uma abordagem não-hierárquica, onde os efeitos não ideais dos transístores (no amplificador e nos interruptores) são tidos em consideração, permitindo a computação precisa da resposta em frequência, mesmo em casos onde o equilíbrio de carga não é atingido antes do final da fase.

Vários exemplos de dimensionamento e optimização são apresentados para demonstrar o desempenho dos métodos propostos. Os protótipos de um filtro de condensadores comutados passa-banda programável de segunda ordem e de um filtro de condensadores comutados rejeita-banda para 50 Hz foram projectados em tecnologia CMOS de 130 nm da UMC e optimizados utilizando o software de optimização proposto com uma tensão de alimentação de 0.9 V. O filtro de condensadores comutados passa-banda tem um consumo de 249 μW. A frequência central do filtro é ajustável entre 3.9 kHz e 7.1 kHz, o ganho entre -6.4 dB e 12.6 dB e o factor de qualidade entre 0.9 e 6.9. Dependendo dos bits de configuração a THD do circuito está entre -54.7 dB e -61.7 dB. O filtro de condensadores comutados rejeita-banda de 50 Hz tem um consumo de 273 μW. A simulação transiente da vista extraída do circuito (C+CC) mostra uma atenuação de 52,3 dB no sinal interferente de 50 Hz e que o sinal de desejado de 5 kHz possui uma THD de -92,3 dB.

Palavras-chave: Algoritmos genéticos, amplificadores de ganho reduzido, desenho assistido por computador, filtros de condensadores comutados, filtro passa-baixo, filtro passa-banda, filtro rejeita-banda, função de custo híbrida, topologia Fleischer-Laker, topologia Sallen-Key.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of Abbreviations and Acronyms

A/D	Analog-to-Digital.
AAF	Anti-Aliasing Filter.
ADC	Analog-to-Digital Converter.
CDB	Charge Division Branch.
CDS	Correlated Double Sampling.
CLG	Closed-Loop Gain.
CMFB	Common-Mode Feedback.
CMOS	Complementary Metal-Oxide-Semiconductor.
DC	Direct Current.
ENOB	Effective Number of Bits.
ESD	Electrostatic Discharge.
FFT	Fast Fourier Transform.
GA	Genetic Algorithm.
GBW	Gain-Bandwidth Product.
GSL	GNU Scientific Library.
IC	Integrated Circuit.
IM3	3rd Order Intermodulation Product.
IoT	Internet-of-Things.
KCL	Kirchhoff's Current Law.
LP	Linear Programming.
LSE	Large-Signal Excitation.
MC	Monte Carlo.

MDAC Multiplying Digital-to-Analog Converter.

MIM Metal Insulator Metal.

MINLP Mixed Integer Nonlinear Programmming.

MNA Modified Nodal Analysis.

MOSFET Metal-Oxide-Semiconductor Field-Effect Transistor.

NMOS n-channel MOSFET.

OP Operating Point.

OSR Oversampling Ratio.

OTA Operational Transconductance Amplifier.

PAC Periodic Alternating Current.

PMOS p-channel MOSFET.

PSS Periodic Steady State.

PVT Process, Voltage supply, and Temperature.

S/H Sample and Hold.

SA Simulated Annealing.

SAR Successive Approximation Register.

SC Switched-Capacitor.

SC-CMFB Switched-Capacitor Common-Mode Feedback.

SCA Switched-Current-Assisting.

SFG Signal Flow Graph.

SNDR Signal-to-Noise + Distortion Ratio.

THD Total Harmonic Distortion.

VCVS Voltage-Controlled Voltage Source.

VGRB Virtual Ground Reference Buffer.

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

The scaling-down of transistors in advanced deep-submicron CMOS technologies has led to undisputed advantages in terms of chip area, transistor transit frequency, and power consumption, mainly exploited by digital circuits. Yet, there are also disadvantages due to the reduction of the transistor's channel length. Ideally, voltages and transistor's dimensions should be scaled proportionally in order to maintain the same electric field. However, the threshold voltage (V_{TH}) can not be scaled in the same proportion as the supply voltage, since a high V_{TH} is needed to achieve a low current when the transistor is turned OFF. With the generalized scaling taken over the past decades, the maximum output swing of analog circuits is decreasing, as well as the intrinsic gain (g_m/g_{ds}) of the transistors, impacting the functionality and performance of analog blocks [1].

Filter circuits are commonly used in data acquisition channels to remove unwanted signals. They can be implemented using passive or active RC filters, g_m-C filters, or switched-capacitor (SC) filters. Typically, these channels require some degree of bandwidth tunability.

In traditional g_m-C or active-RC filters architectures, the frequency tuning is achieved through the use of on-chip capacitor and resistor banks, increasing the circuit's cost. Although their speed-to-power efficiency has improved with the scaling of CMOS technology, these circuits have become harder to design due to the reduction of the supply voltage and of the intrinsic gain of the transistors.

SC filters, on the other hand, can be a competitive alternative for their inherent frequency tuning, by changing the clock's frequency; for their small die area, when compared with other types of filters (especially for low frequency filters); and for their low sensitivity to parameter deviation, since the circuit's transfer function is given by capacitor ratios, which have a very good accuracy in integrated circuit (IC) technology, and not by their absolute value. However, traditional active SC filters require high-gain high-bandwidth opamps, which are harder to design in advanced technologies.

In the case of SC filters, to overcome the difficulty of designing high-gain high-bandwidth opamps, alternative architectures can be used, where the opamp is replaced with a voltage buffer or a low-gain (< 2) amplifier. Although this simplifies the amplifier's design, it also eliminates the filter's virtual ground node. This means that, even if parasitic-insensitive switched networks are used, the circuit's transfer function is susceptible to the effects of parasitic capacitances. However, the historical disadvantage of parasitic-sensitive switched networks (parallel branch) is no longer critical, since the scaling down of transistors and supply voltage has also led to a reduction of the parasitic capacitances, allowing their influence to be compensated during the circuit's design process [2]. The use of parasitic-sensitive networks has the advantage of allowing the implementation of filters with more than one pole per amplifier, improving the filter's power-per-pole efficiency. Additionally, due to the amplifier's low gain, it is possible to achieve a higher bandwidth for the same power, when compared with implementations using amplifiers with a high gain.

Independently of the type of amplifier used, the manual design of SC filters can be a challenging and time consuming process, since the value of a single design variable can change several performance parameters (cutoff frequency, gain, poles quality factor). Even after finding a set of parameter values that produce the desired filtering function, it is also necessary to evaluate the solution's sensitivity to process, voltage supply, and temperature (PVT) variations and mismatches errors, to improve the circuit's yield after fabrication. As a consequence, the number of evaluations increase considerably.

As an alternative to the manual design of SC filters, optimization tools can be used to automatize the circuit's design process. The frequency response of a SC filter can be obtained by simulating the circuit's impulse response. However, due to the large computation time needed to get enough points to accurately compute the frequency response, this approach in not efficient to be used in an optimization environment, due to the number of evaluation needed to find an optimal solution. A more efficient method is then needed to obtain an optimized solution, with good accuracy, within a reasonable amount of computation time.

The main motivation of this PhD work is the development of an efficient analysis and optimization software for the design of SC filters, capable of accurately obtain and optimize the filter's frequency response, considering the non-ideal effects of the circuit's transistors, within a reasonable amount of computation time.

The proposed tool has been used to design and optimize the prototypes of a programmable bandpass SC biquad filter, implemented with a low-gain amplifier, and the prototype of a 50 Hz notch SC filter, implemented with high-gain amplifiers. These circuits have been designed in a standard 130 nm CMOS technology with a supply voltage of 0.9 V. Measurement results of the bandpass SC filter show that the filter's central frequency can be tuned between 3.9 kHz and 7.1 kHz, the gain between -6.4 dB and 12.6 dB, and the quality factor between 0.9 and 6.9. Depending on the bit configuration of the capacitor banks, the THD is between -54.7 dB and -61.7 dB. The transient simulation of the extracted view (C+CC) of the notch SC filter shows an attenuation of 52.3 dB in the 50 Hz interference and that the desired 5 kHz signal has a THD of -92.3 dB. The filters have a total power consumption of 249 μW and 273 μW, respectively.

1.2 Research Question and Hypothesis

How to design highly selective SC filter circuits in nanometer CMOS technologies using the available transistors with intrinsic low-gain while reducing the power consumption and area? Is it possible to achieve these requirements while maintaining insensitivity to PVT variations and mismatch errors?

If low DC gain amplifiers can be easily designed in nanometer technologies and it is possible to use SC filter topologies with low-gain amplifiers then, it should be possible to design high performance, low power, low area filters in nanometer CMOS technologies.

If a numerical methodology for the analysis of SC filters, capable of reliably describing the circuit's behavior and accounting for the non-ideal effects of the circuit's transistors can be developed then, it should be possible to implement an optimization software, using said methodology, capable of automatically finding the best filter design, for a given set of filter specifications and, at the same, that is robust to PVT variations and mismatch errors.

1.3 Original Contributions and Publications

The main contributions of this PhD thesis are as follows:

- Development of an efficient numerical methodology to obtain the frequency response of SC filters, based on the circuit's first-order differential equations. This methodology uses a non-hierarchical approach, where the non-ideal effects of the transistors in the amplifier and in the switches are taken into consideration, using the transistors' medium frequency small-signal model, allowing the accurate computation of the circuit's frequency response, at transistor level, even in the case of incomplete settling in the SC branches. Since the analysis methodology is purely numeric in nature, it can be efficiently implemented in a computer. The result of this work has been published in [3].
- Development of a software library for the optimization of SC filters using hybrid cost functions (equation-based/simulation-based). This library was integrated into an existing circuit optimization software platform, based on the open-source circuit simulator Ngspice. Initially, equations are used in the cost function to estimate the filter's frequency response. This reduces the computation time of each evaluation, allowing the use of large populations to increase the probability of completely exploring the design space. Once all specifications are met, the population size is reduced and transient simulations of the circuit's impulse response are used in the cost function, resulting in the accurate determination of the filter's frequency response and allowing to accurately compensate the parasitic capacitances, which effect is harder to estimate through equations. This approach allows the final design solution to be obtained in a reasonable amount of computation time. The result of this work has been published in [4]–[6].
- Implementation of the prototype of a programmable second-order bandpass SC filter using low gain amplifiers and of a 50 Hz notch SC filter using high gain amplifiers in UMC 130 nm CMOS technology, with a supply voltage of 0.9 V, using the proposed analysis and optimization methodologies.

The research work performed during the PhD resulted in the following authored and co-authored publications:

- H. Serra, R. Santos-Tavares, and N. Paulino, "A top-down optimization methodology for SC filter circuit design using varying goal specifications", in *Proc. Doctoral Conf. Computing, Elect., Ind. Syst. (DoCEIS)*, Apr. 2014, pp. 535–542. DOI: 10.1007/978-3-642-54734-8_59.
- H. Serra, R. Santos-Tavares, and N. Paulino, "A top-down optimization methodology for SC filter circuit design", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2014, pp. 1672–1675. DOI: 10.1109/ISCAS.2014.6865474.
- H. Serra, R. Madeira, and N. Paulino, "Analysis of a multi-ratio switched capacitor DC-DC converter for a supercapacitor power supply", in *Proc. Doctoral Conf. Computing, Elect., Ind. Syst. (DoCEIS)*, Apr. 2015, pp. 477–485. DOI: 10.1007/978-3-319-16766-4_51.
- H. Serra, R. Santos-Tavares, and J. Goes, "Automatic design of high-order SC filter circuits", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 1937–1940. DOI: 10.1109/ISCAS.2015.7169052.
- H. Serra, R. Santos-Tavares, and N. Paulino, "A numerical methodology for the analysis of switched-capacitor filters taking into account non-ideal effects of switches and amplifiers", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 1, pp. 61–71, Jan. 2017. DOI: 10.1109/TCSI.2016.2601343.
- N. Pereira, H. Serra, and J. Goes, "A two-step radio receiver architecture fully embedded into a charge-sharing SAR ADC", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2017, pp. 1201–1204. DOI: 10.1109/ISCAS.2017.8050563.
- H. Serra, J. P. Oliveira, and N. Paulino, "A 50 Hz SC notch filter for IoT applications", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2017, pp. 2435–2438. DOI: 10.1109/ISCAS.2017.8050904.

1.4 Thesis Organization

Besides this introductory chapter, the thesis is organized in seven additional chapters, which are briefly described below.

In Chapter 2, an overview of the fundamental concepts for SC filters and their main building blocks is given, including non-ideal effects that need to be considered when designing this type of circuits. Different SC resistor emulation networks are shown and two SC integrators, one using a parasitic-sensitive network and the other using a parasitic-insensitive network, are described to show the advantage of using parasitic-insensitive switched networks to minimize the effects of the parasitic capacitances when an amplifier with high gain is used.

In Chapter 3, an overview of design techniques used in SC filters to reduce the capacitance spread and to improve the trade-off of speed and linearity with the circuit's power consumption is given. Four techniques to reduce a SC filter's capacitance spread are described, using a SC integrator as an example. The capacitance spread is improved by reducing the amount of charge delivered from the sampling capacitor to the integrating capacitor or by integrating only a small amount of charge, while destroying the rest. A switched-current-assisting (SCA) technique for a SC integrator is also described, which improves the trade-off of speed and linearity with the circuit's power consumption. This is achieved by using an assisting branch connected to the circuit's output to alleviate the slew rate and bandwidth requirements of the main amplifier. A virtual ground reference buffer (VGRB) technique is described where the amplifier's feedback factor is improved, without affecting the signal gain, through the use of an auxiliary voltage buffer, breaking away from the inverse relation between signal gain and feedback factor, relaxing the amplifiers bandwidth and noise requirements. A recycling SC-buffer biquad is described. Since this circuit is implemented without gain, effects of non-linearities and parasitic capacitances are intrinsically low, making it easier to achieve higher bandwidths for the same power when compared with SC filters implemented with high-gain amplifiers.

In Chapter 4, an overview of analysis methods for SC filters is given. Several analysis and optimization software programs are briefly described. A general method for the analysis of ideal switched networks, based on matrix computations, is described. Three approaches to deal with the minimization of the total capacitance value in SC circuits are described. One of the methods uses a mixed integer nonlinear programmming (MINLP) software to find the optimal capacitance sizing for a given prototype transfer function, while minimizing the number of capacitors needed and prioritizing solutions with lower sensitivity to component mismatches. The other two methods use linear programming (LP) and simulated annealing (SA) algorithms to minimize the total capacitance value. Both methods are compared to show their capability in finding global minimums for the optimal capacitance sizing problem.

In Chapter 5, two methods for the design of SC filters are proposed. In the first method, the SC filter is designed hierarchically, i.e., the filter's transfer function is initially designed considering every component as ideal and, afterwards, the amplifier circuit and the switches are designed with the necessary performance to replace the ideal components and the parasitic capacitances are compensated into the ideal capacitance values to obtain a frequency response that is as close as possible to the one obtained from the ideal transfer function. In the second method, the SC filter is designed using a non-hierarchical approach, considering the non-ideal effects of the circuit's transistors, allowing the accurate computation of the filter's frequency response. Some design examples are given in this chapter to demonstrate the accuracy of both methods.

In Chapter 6, an optimization software, using genetic algorithms (GAs) and two equation-based optimization methodologies, for SC filters, is described. The first optimization methodology (hierarchical) optimizes the filter's frequency response using a multi-step approach where, initially, the filter's ideal transfer function is optimized and, once the specifications are met, the next two steps

are used to optimize the amplifier's closed-loop transfer function and the switches RC time constants to replace the ideal blocks used in the first step. The final step is used to validate the equation-based design, through transient simulations of the impulse response, and to accurately compensate the effect parasitic capacitances. The second optimization methodology (non-hierarchical) optimizes the filter's frequency response using a single system of equations that completely describe the circuit's continuous-time behavior in any clock phase, independently of the number of phases that control the circuit. Once a good solution is found, it is validated through a transient simulation of the impulse response. Both optimization methods include PVT and Monte Carlo (MC) optimization modes to increase the solution's robustness under different PVT corners and to mismatch variations. Several optimization examples are given in this chapter to demonstrate the performance of both optimization methodologies, including an example where the optimization times of both methods are compared.

In Chapter 7, the prototypes of a programmable bandpass SC biquad filter, using a low-gain amplifier, and of a 50 Hz notch SC filter, using high-gain amplifiers, are presented. Both circuits are implemented with 0.9 V supply voltage in a 130 nm CMOS technology and designed using the analysis and optimization methodologies described in Chapters 5 and 6. A comparison between the programmable bandpass SC filter's frequency responses obtained from the schematic and post-layout simulations and from the measurement of the test board is given. In the case of the 50 Hz notch SC filter, the comparison is made between the schematic and post-layout simulations.

Finally, in Chapter 8, the conclusions are drawn and the future work is discussed.

CHAPTER 2

BACKGROUND ON SWITCHED-CAPACITOR FILTERS

Interest in switched-capacitor (SC) networks began in the late 1970's due to the possibility of implementing precision analog filters using integrated circuit (IC) technology, since it is possible to achieve good accuracy in the ratio between two capacitor values. Typically, high valued resistors are used in the implementation of analog filters, that can be replaced by small on-chip capacitors, making it an attractive option for monolithic fabrication, since it occupies a small substrate area.

SC circuits operate as analog discrete-time circuits. When used as filters, these circuits have an accurate frequency response, good linearity, and good dynamic range. The accuracy of these circuits results from the time constants being determined by capacitor ratios, that typically have an accuracy close to 0.1 %, while integrated RC circuits have a time constant error that can range from 20 to 50 %, due to the accuracy of the resistors and the capacitors. Another advantage of these circuits is the incorporation of a certain degree of frequency tuning, that is achieved by changing the circuit's clock frequency [8].

There are also several non-ideal characteristics of SC circuits which need to be considered, such as charge injection, offset error, noise, and parasitic capacitances [9]. Generally, due to the sampling of the signal, SC filters need an anti-aliasing filter at the input and a smoothing filter at the output [10].

This chapter provides a brief overview on the building blocks of SC filters, the non-ideal effects of these blocks, that need to be carefully considered during the circuit's design, and some techniques commonly used to minimize these effects.

2.1 Switched-Capacitor Filters Building Blocks

SC filters are implemented using switches, capacitors, non-overlapping clock generators, and opamps. These blocks are briefly described in this section along with some techniques to eliminate or minimize their non-linear effects.

2.1.1 Switches

In SC circuits, switches need high resistance when they are open, in order to minimize charge leakage, and low resistance when they are closed, to ensure that charge equilibrium is achieved within the time the switch is closed. Depending on their size, MOSFET transistors satisfy these requirements, having very high OFF resistance and low ON resistance ($G\Omega$ and $k\Omega$, respectively). These values change inversely with the width of the transistor and directly with the length.

Fig. 2.1 shows three possible configurations for the implementation of the switches: NMOS transistors, which are better for lower input voltages since they stop conducting when the input voltage is close to $V_{DD} - V_{tn}$; PMOS transistors, which are better for higher input voltages since they only conduct when the voltage is above $|V_{tp}|$; and transmission gates, which allow rail-to-rail input signal swing. Lower voltages will travel via the NMOS switch, since it offers less resistance to the signal while, for the same reason, higher voltages will travel via the PMOS switch.

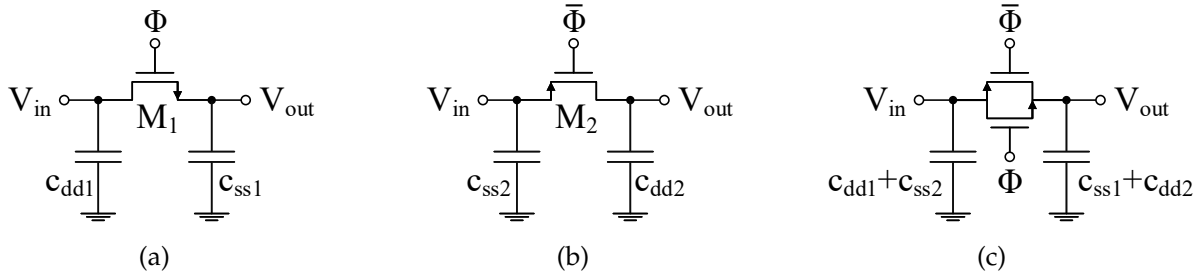


Figure 2.1: MOS switches: (a) NMOS, (b) PMOS, (c) Transmission gate.

When designing parasitic-sensitive SC filters, it is necessary to consider the effect of parasitic capacitances (c_{dd} and c_{ss}) since the filter's transfer function is affected by these capacitances. Although the resistance of a switch can be decreased by increasing the transistors width, in order to guarantee charge equilibrium before the switch opens, doing so will increase the value of the parasitic capacitances. It is important to note that the parasitic capacitances of MOS transistors are non-linear, i.e., their value changes with the applied voltage in the transistor's terminals and depending on if the switch is open or closed.

Fig. 2.2 shows an example of the switches' ON resistance and parasitic capacitances as a function of the input voltage (1:5 ratio between the NMOS and the PMOS devices). Although the transmission gate allows rail-to-rail signal swing and a more linear resistance value within the output voltage range, it takes up more area, requires complementary phases, and has larger parasitic capacitances. In high speed signals, both switches need to turn OFF simultaneously otherwise the distortion in the signal will increase.

Decreasing the ON resistance leads to larger switches and bigger parasitic capacitances. In these situations, and especially in parasitic-sensitive circuits, clock bootstrapping can be used to increase the voltage applied to the gate of the transistor, decreasing the resistance value and increasing its linearity, allowing the use of smaller switches and a single NMOS device instead of a transmission gate, also decreasing the value of the parasitic capacitances [11].

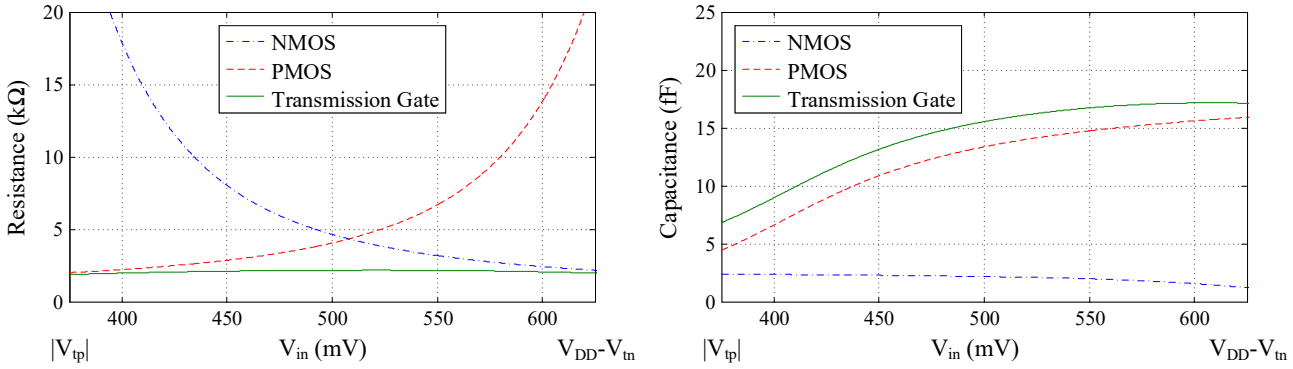


Figure 2.2: Example of switch (a) r_{ds} resistance and (b) c_{ss} capacitance, as a function of the V_{in} voltage.

Besides the non-linear effects of the resistance and parasitic capacitances, switches are susceptible to other non-ideal effects, such as charge injection and clock feedthrough. Both of these effects produce disturbances in the sampled voltage when the transistor is turned OFF. Charge injection occurs due to charge being released from the transistor's channel to the drain and source junctions and clock feedthrough due to the coupling between the gate-source and gate-drain overlap capacitances [12].

Frequently used techniques to improve charge injection include the use differential signals, canceling offset errors and even-order distortion, although gain errors and odd-order distortion remain.

Another technique is shown in Fig. 2.3, using a dummy switch configuration to perform charge injection cancellation. When the main transistor (M_1) turns OFF and the dummy (M_2) turns ON, the charge released by the former is absorbed by the latter to form the channel. Typically the dummy transistor is designed to have half the width of the main transistor to cancel charge injection [13]. Note that this technique is sensitive to the alignment of both clock phases.

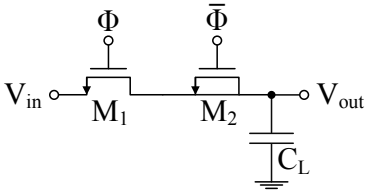


Figure 2.3: Simple circuit using dummy switch technique.

Fig. 2.4 shows yet another technique, the bottom plate sampling. The bottom transistor (M_2) turns OFF sightly before the top transistor (M_1), injecting a constant amount of charge that can be eliminated using differential signaling. When the top transistor turns OFF, the bottom plate of capacitor C_L is floating and no current will flow through the capacitor, as a result no signal dependent charge is injected [14].

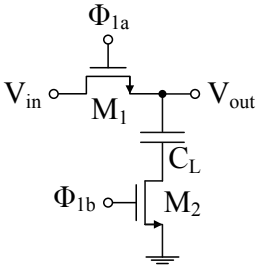
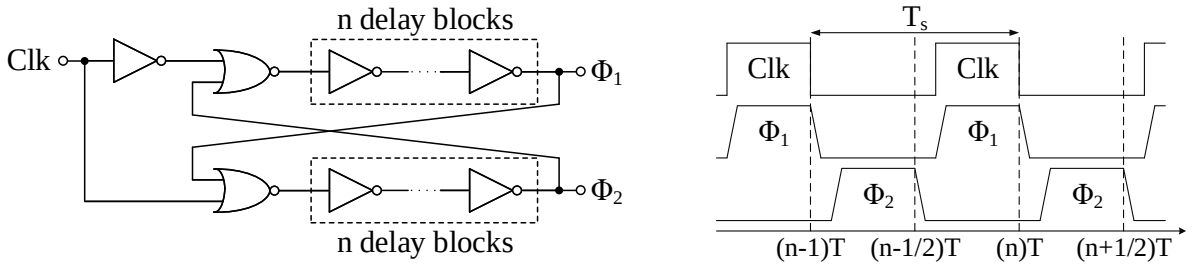


Figure 2.4: Simple circuit using bottom plate sampling technique.

2.1.2 Non-Overlapping Clock Phases

In SC filters, switches require at least a pair of non-overlapping clock phases to perform the charge transfer. Typically, the phases do not overlap, in order to ensure that no charge is accidentally lost by having switches with different phases closed at the same time. Fig. 2.5 shows an example of a two-phase non-overlapping clock generator [15].



(a) (b)

Figure 2.5: Two-phase non-overlapping clock generator: (a) Circuit implementation, (b) Phase scheme.

2.1.3 Capacitors

Depending on the network used to emulate resistors and the SC filter's architecture, capacitors are another element where it may be necessary to consider the effects of parasitics. The bottom plate parasitic capacitance can be as high as 20 % of the nominal value, while the top plate can be as high as 5 %, depending on the technology and capacitor structure [8].

2.1.4 Operational Amplifiers

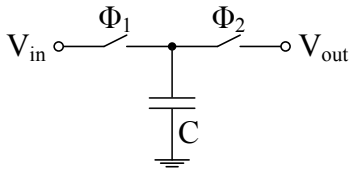
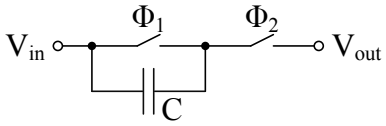
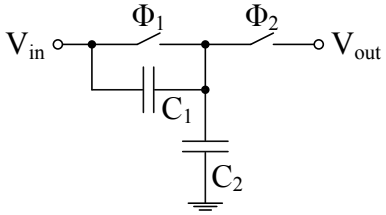
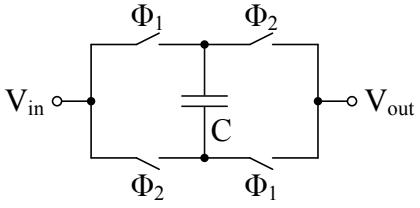
Depending on the amplifier's gain, opamps can provide a virtual ground node in SC circuits. When using parasitic-insensitive networks, parasitic capacitances connected to this node do not influence the circuit's transfer function since, in one phase, they will be connected to the circuit's ground, and to the virtual ground during the other. Note, however, that the parasitic capacitances will still influence the time constant of the nodes they are connected to. Opamps also have non-ideal effects that influence the performance of SC circuits like, DC gain, unity gain frequency and phase margin, slew rate, and common-mode voltage [16].

In [17], a general technique to reduce the amplifier's finite gain effect is described. The basic principle of this technique is to obtain an estimate of the amplifier's finite gain error during the first phase and subsequently use this error for correction during the second phase. Since the input has to be held constant during these two phases, a third phase might be necessary when this circuit is connected to other stages. With this approach, it was reported that the finite gain error, which is proportional to $1/A$ in conventional circuits [18], becomes proportional to $1/A^2$, where A is the amplifier's gain.

2.2 Switched-Capacitor Resistor Emulation Networks

SC circuits emulate resistors using switches, switching periodically at a given frequency, and capacitors. Table 2.1 shows four different switched networks that emulate resistors, their equivalent resistance, and the charge stored in the capacitors after each phase [19].

Table 2.1: SC resistor emulation circuits.

Circuit	Schematic	R_{eq}	$Q(\Phi_1)$	$Q(\Phi_2)$
Parallel		$\frac{T}{C}$	$V_{in}\ C$	$V_{out}\ C$
Series		$\frac{T}{C}$	0	$(V_{in} - V_{out})\ C$
Series-Parallel		$\frac{T}{C_1 + C_2}$	0 $V_{in}\ C_2$	$(V_{in} - V_{out})\ C_1$ $V_{out}\ C_2$
Bilinear		$\frac{1}{4}\frac{T}{C}$	$(V_{in} - V_{out})\ C$	$(V_{out} - V_{in})\ C$

Considering the series-parallel network and the phase scheme shown in Fig. 2.6, in which the input current flows into the circuit in both phases, the calculation of the average current contemplates both phases.

$$i_{avg} = \frac{1}{T}\left(\int_0^{T/2} \mathrm{d}q_{C_2}(t) + \int_{T/2}^{T} \mathrm{d}q_{C_1}(t)\right) = \frac{Q_{C_2}(T/2) - Q_{C_2}(0)}{T} + \frac{Q_{C_1}(T) - Q_{C_1}(T/2)}{T} \tag{2.1}$$

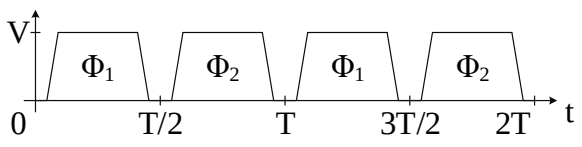


Figure 2.6: Non-overlapping clock phase scheme.

Replacing the charge variables with the corresponding values that are shown in Table 2.1,

$$i_{avg} = \frac{(V_{in} - V_{out})C_2}{T} + \frac{(V_{in} - V_{out})C_1 - 0}{T} \tag{2.2}$$

and considering that the average current that flows through the resistance,

$$i_{avg} = \frac{V_{in} - V_{out}}{R} \tag{2.3}$$

by equating (2.2) and (2.3), the equivalent resistance for the series-parallel network is obtained.

$$R_{eq} = \frac{T}{C_1 + C_2} \tag{2.4}$$

Depending on the switched network used and the topology of the filter, it may be possible to implement parasitic-insensitive circuits. The parasitic-sensitive and parasitic-insensitive integrators are briefly described next.

2.2.1 Parasitic-Sensitive Switched-Capacitor Integrator

The parasitic-sensitive SC integrator, shown in Fig. 2.7, was presented in [20] by replacing the resistor in a Miller integrator with a T (parallel) switched network.

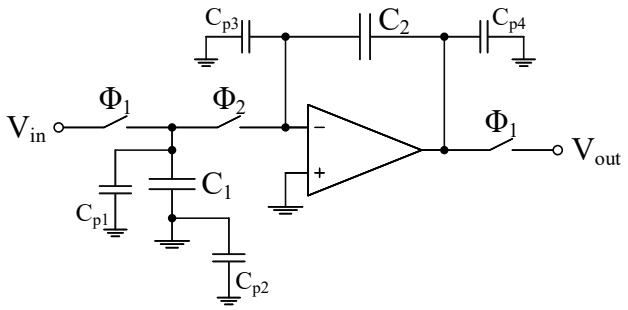


Figure 2.7: Parasitic-sensitive switched-capacitor integrator.

Considering the amplifier gain as infinity, analyzing the charge across the capacitors in each phase, and considering that the signal will be sampled at the end of phase Φ_1, Table 2.2 is obtained. The capacitances C_{p1-p4} represent the equivalent parasitic capacitances from the switches, opamp, and capacitors, associated to their respective node.

Table 2.2: Capacitor charge in each phase of the parasitic-sensitive integrator.

	Φ_1	Φ_2	Φ_1
Q_{C_1}	$V_{in}[(n-1)T]\ C_1$	≈ 0	$V_{in}[nT]\ C_1$
Q_{C_2}	$-V_{out}[(n-1)T]\ C_2$	$-V_{out}[(n-1/2)T]\ C_2$	$-V_{out}[nT]\ C_2$
$Q_{C_{p1}}$	$V_{in}[(n-1)T]\ C_{p1}$	≈ 0	$V_{in}[nT]\ C_{p1}$
$Q_{C_{p2}}$	0	0	0
$Q_{C_{p3}}$	≈ 0	≈ 0	≈ 0
$Q_{C_{p4}}$	$V_{out}[(n-1)T]\ C_{p4}$	$V_{out}[(n-1/2)T]\ C_{p4}$	$V_{out}[nT]\ C_{p4}$

Considering the transitions ($\Phi_1 \rightarrow \Phi_2$) and ($\Phi_2 \rightarrow \Phi_1$), by adding all the capacitors that are connected to the virtual ground node at the end of that transition (Φ_2 in the first case and Φ_1 in the second), i.e., the node where charge conservation occurs, (2.5) is obtained.

$$\begin{aligned} &\Phi_1 \rightarrow \Phi_2 : V_{in}[(n-1)T](C_1 + C_{p1}) - V_{out}[(n-1)T]\, C_2 = -V_{out}[(n-1/2)T]\, C_2 \\ &\Phi_2 \rightarrow \Phi_1 : -V_{out}[(n-1/2)T]\, C_2 = -V_{out}[nT]\, C_2 \end{aligned} \tag{2.5}$$

Combining both equations in (2.5) and applying the $\mathcal{Z}$-transform, the parasitic-sensitive integrator transfer function (2.6) is obtained.

$$H(z) = \frac{V_{out}}{V_{in}} = -\left(\frac{C_1 + C_{p1}}{C_2}\right)\left(\frac{z^{-1}}{1 - z^{-1}}\right) \tag{2.6}$$

Considering the effects of the parasitic capacitances, this circuit's gain coefficient is dependent on C_{p1}. Capacitance C_{p2} and C_{p3} do not influence the transfer function since they are connected to ground and to the virtual ground, respectively, and capacitance C_{p4} since it is driven by the circuit's output.

2.2.2 Parasitic-Insensitive Switched-Capacitor Integrator

To overcome the non-linear effect of the parasitic capacitance C_{p1}, new parasitic-insensitive structures were developed [21]. One of these structures is shown in Fig. 2.8, where the T switched network is replaced with a π switched network.

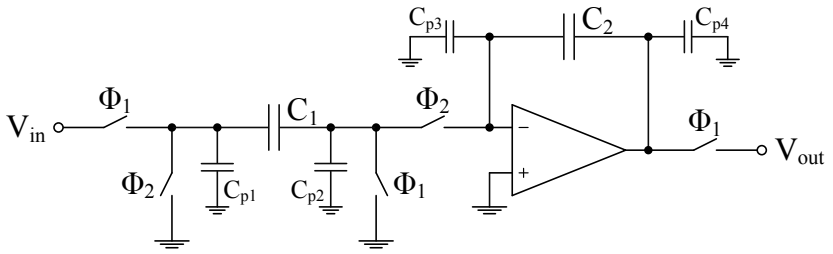


Figure 2.8: Parasitic-insensitive switched-capacitor integrator.

Considering the amplifier gain as infinity, analyzing the charge across the capacitors in each phase, and considering that the signal will again be sampled at the end of phase Φ_1, Table 2.3 is obtained. The capacitances C_{p1-p4} represent the equivalent parasitic capacitances from the switches, opamp, and capacitors, associated to their respective node.

Table 2.3: Capacitor charge in each phase of the parasitic-insensitive integrator.

	Φ_1	Φ_2	Φ_1
Q_{C_1}	$-V_{in}[(n-1)T]\, C_1$	≈ 0	$-V_{in}[nT]\, C_1$
Q_{C_2}	$-V_{out}[(n-1)T]\, C_2$	$-V_{out}[(n-1/2)T]\, C_2$	$-V_{out}[nT]\, C_2$
$Q_{C_{p1}}$	$V_{in}[(n-1)T]\, C_{p1}$	0	$V_{in}[nT]\, C_{p1}$
$Q_{C_{p2}}$	0	≈ 0	0
$Q_{C_{p3}}$	≈ 0	≈ 0	≈ 0
$Q_{C_{p4}}$	$V_{out}[(n-1)T]\, C_{p4}$	$V_{out}[(n-1/2)T]\, C_{p4}$	$V_{out}[nT]\, C_{p4}$

Considering the transitions ($\Phi_1 \rightarrow \Phi_2$) and ($\Phi_2 \rightarrow \Phi_1$), (2.7) is obtained by adding all the capacitors that are connected to the virtual ground node at the end of that transition (Φ_2 in the first case and Φ_1 in the second).

$$\begin{aligned} \Phi_1 \rightarrow \Phi_2 &: -V_{in}[(n-1)T]\,C_1 - V_{out}[(n-1)T]\,C_2 = -V_{out}[(n-1/2)T]\,C_2 \\ \Phi_2 \rightarrow \Phi_1 &: -V_{out}[(n-1/2)T]\,C_2 = -V_{out}[nT]\,C_2 \end{aligned} \tag{2.7}$$

Combining both equations in (2.7) and applying the $\mathcal{Z}$-transform, the parasitic-insensitive integrator transfer function (2.8) is obtained.

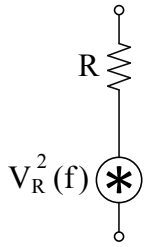
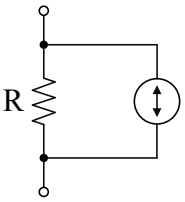
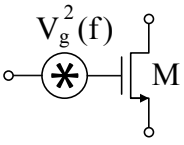
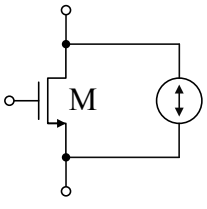
$$H(z) = \frac{V_{out}}{V_{in}} = \left(\frac{C_1}{C_2}\right)\left(\frac{z^{-1}}{1-z^{-1}}\right) \tag{2.8}$$

Changing the resistor emulation network from a branch in T to a branch in π, the parasitic capacitances in this circuit no longer influence the circuit's transfer function since they are: connected to the virtual ground (C_{p2} and C_{p3}) or to the circuit's ground (C_{p2}); being driven by the circuits output (C_{p4}); or in the case of C_{p1}, being charged by the input in one phase and discharged to ground in the next. As long as the time constant is long enough to ensure charge equilibrium before the end of the clock phase, C_{p1} will not influence the performance of the circuit.

2.3 Noise in Switched-Capacitor Circuits

Although capacitors do not generate noise, SC circuits are not noise-free because transistors, in any operating region, introduce noise. Table 2.4 shows the noise models for resistors (transistors in the linear region) and for transistors in the saturation region.

Table 2.4: Circuit elements and noise models.

Element	Noise Models	
Resistor	 $V_R^2(f) = 4kTR$	 $I_R^2(f) = \frac{4kT}{R}$
MOS transistor (saturation region)	 $V_g^2(f) = \frac{k}{WLC_{ox}f}$	 $I_d^2(f) = 4kT\gamma g_m$

Transistors in the linear region (switches) exhibit thermal noise that can be modeled by a voltage source, with a white noise spectral density of $4kTR$, in series with a noiseless resistor. In the saturation region the dominant noise sources are thermal and flicker noise. In this region the channel is no longer homogeneous and exhibits pinching near the drain. In this region, the channel's resistance increases by a factor of γ^{-1} (3/2 for long channel devices), i.e., $R_c = (\gamma WL^{-1}\mu_n C_{ox}(V_{GS} - V_T))^{-1}$. The noise current in the drain, considering that $g_m = WL^{-1}\mu_n C_{ox}(V_{GS} - V_T)$, is then given by $4kT\gamma g_m$ [8].

Flicker can be modeled as a voltage source in series with the gate of the transistor and is inversely proportional to the transistor's area. In low frequency applications, flicker noise ($1/f$) needs to be carefully considered because it dominates over other noise types. Although increasing the transistor's area decreases flicker noise, it is not a desirable solution since it increases the area and cost of the circuit.

In SC circuits, three techniques are commonly used to reduce flicker noise: Chopping, by feeding the circuit's input signal with different signs every clock period, to modulate flicker to a frequency outside the signal band; large-signal excitation (LSE), since flicker depends not only on the present bias state of the transistor but also on its bias history, the transistors are switched OFF temporarily in order to attempt to clear its memory; and correlated double sampling (CDS), since flicker has a flat auto-correlation function, subtracting two correlated samples allows the removal of the undesired offset [22].

Unlike continuous-time circuits, which are under constant bias conditions, SC circuits change configuration at every clock phase therefore, the noise analysis is done for every phase. Fig. 2.9 shows the equivalent noise circuits in each clock phase of the parasitic-insensitive integrator shown in Fig. 2.8, neglecting the parasitic capacitances.

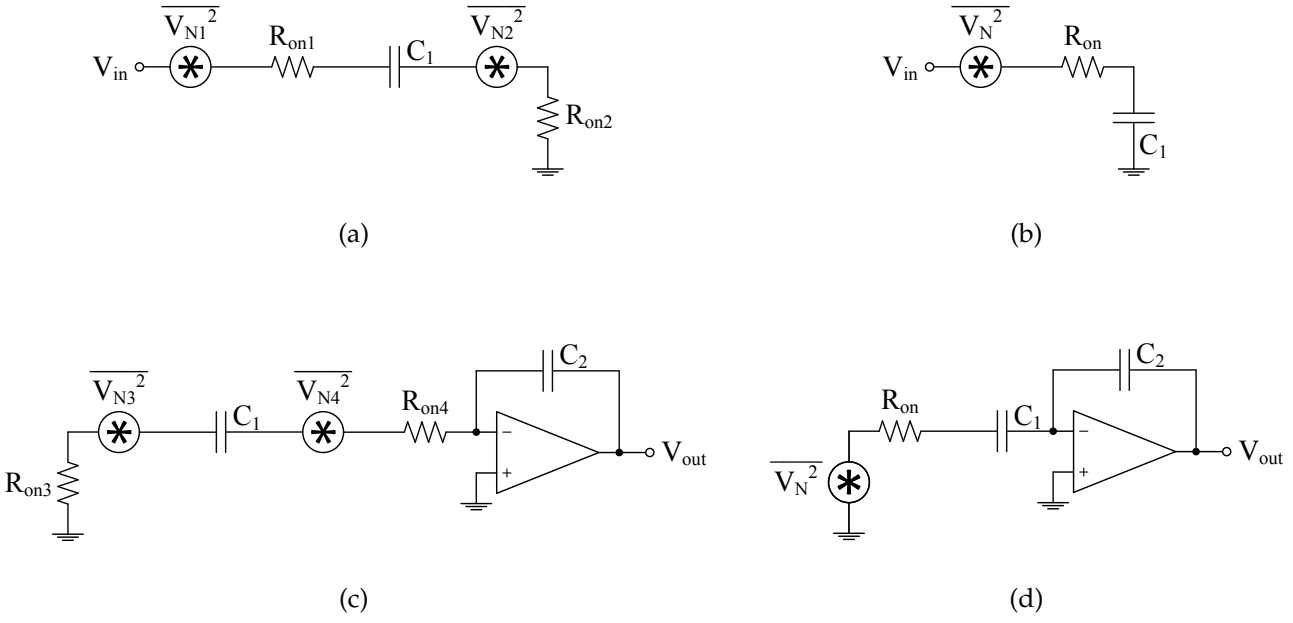


Figure 2.9: Equivalent noise circuits during phase: (a) Φ_1, (b) Φ_1 (simplified), (c) Φ_2, (d) Φ_2 (simplified).

During phase Φ_1, the two switches introduce noise which will add an error in the charge being stored in capacitor C_1. During this phase the equivalent circuit is a simple RC (Fig. 2.9(b)) where the noise stored in C_1 (2.9) is calculated by integrating the switches noise spectral density multiplied by the square of the absolute value of the circuit's transfer function. When integrated to infinity, the noise is independent of the resistance value since, although the noise increases with larger resistances, the filtering also increases, canceling the effect of the resistance value. In (2.9), k represents the Boltzmann constant and T the temperature (in Kelvin).

$$\begin{aligned}\overline{V_N{}^2}(\Phi_1) &= \int_0^\infty \left(\frac{\overline{V_{N_1}{}^2}}{\Delta f}(f) + \frac{\overline{V_{N_2}{}^2}}{\Delta f}(f) \right) |H^{\Phi_1}(f)|^2 \, df \\ &= \int_0^\infty 4kT(R_{on_1} + R_{on_2}) \left| \frac{1}{1 + 2\pi f C_1 (R_{on_1} + R_{on_2}) j} \right|^2 df \\ &= \frac{kT}{C_1}\end{aligned} \tag{2.9}$$

The noise generated during phase Φ_2 (2.10), neglecting the noise generated by the amplifier, can be calculated in the same manner using the equivalent circuit's (Fig. 2.9(d)) transfer function during this phase.

$$\begin{aligned}\overline{V_N{}^2}(\Phi_2) &= \int_0^\infty \left(\frac{\overline{V_{N_3}{}^2}}{\Delta f}(f) + \frac{\overline{V_{N_4}{}^2}}{\Delta f}(f) \right) |H^{\Phi_2}(f)|^2 \, df \\ &= \int_0^\infty 4kT(R_{on_3} + R_{on_4}) \left| -\frac{C_1}{C_2} \, \frac{1}{1 + 2\pi f C_1 (R_{on_3} + R_{on_4}) j} \right|^2 df \\ &= \left(\frac{C_1}{C_2} \right)^2 \frac{kT}{C_1}\end{aligned} \tag{2.10}$$

The total noise power at the output (2.11) is the sum of the noise power in both phases. Since the noise stored in C_1 during phase Φ_1 will be amplified in phase Φ_2, it needs to be multiplied by the square of the gain factor (C_1/C_2).

$$\overline{V_{out_N}{}^2} = \left(\frac{C_1}{C_2} \right)^2 \overline{V_N{}^2}(\Phi_1) + \overline{V_N{}^2}(\Phi_2) \tag{2.11}$$

CHAPTER 3

DESIGN TECHNIQUES TO INCREASE PERFORMANCE OF SWITCHED-CAPACITOR FILTERS

Due to the switching nature of switched-capacitor (SC) circuits, an anti-aliasing filter (AAF) needs to be used, before the sampling of the signal, to remove signals above half the sampling frequency (F_s) since, after sampling, these signals will be folded back into the baseband ($f < F_s/2$), at intervals of $F_s/2$, limiting the system's performance. To simplify the specifications requirements of the AAF and allow the use of a passive structure, it is convenient to use a higher sampling frequency, however, doing so increases the SC filter's capacitance spread and the bandwidth (power consumption) requirements of the amplifiers.

This chapter presents some design techniques to improve the SC filter's capacitance spread and the trade-off of speed and linearity with the circuit's power consumption.

3.1 Active SC Filters using Capacitance Spread Reduction Techniques

In this section, four capacitance spread reduction techniques are described. To have a base of comparison between each technique, the switched networks, used to improve the circuit's capacitance spread, are tested for the same circuit topology (SC integrator) and their transfer functions are designed to result in the same frequency response.

The basic concept of these networks is to reduce the amount of charge that is delivered to the integrating capacitor, making its capacitance seem larger than it actually is, allowing the use of a smaller sized capacitor, resulting in a reduction of the capacitance spread.

3.1.1 Basic SC Integrator

The SC integrator topology [23], shown in Fig. 3.1, is used to have a base of comparison for the capacitance spread reduction techniques described in this section.

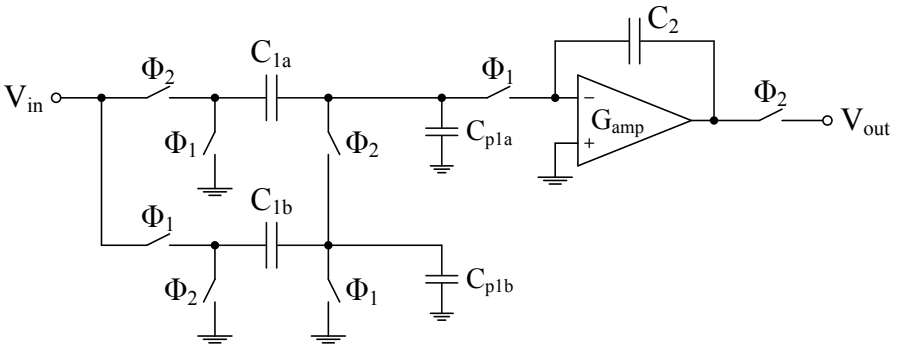


Figure 3.1: Basic SC integrator.

Considering that the signal is sampled at the end of clock phase Φ_2, using the principle of charge conservation, the SC integrator discrete-time transfer function is given by (3.1), where the gain of the amplifier is considered as infinity.

$$H_{basic}^{\Phi_2}(z) = \frac{C_{1a}(C_{1b} + C_{p1b}) + C_{1b}(C_{1a} + C_{p1a})\, z^{-1/2}}{C_2(C_{1a} + C_{1b} + C_{p1a} + C_{p1b})(z-1)} \tag{3.1}$$

Assuming that $C_{1a} = C_{1b} = C_1$ and neglecting the influence of the parasitic capacitances (C_{p1a}, C_{p1b}), equation (3.1) can be simplified to:

$$H_{basic}^{\Phi_2}(z) = \frac{C_1}{2\,C_2}\left(\frac{1 + z^{-1/2}}{z-1}\right) \tag{3.2}$$

3.1.2 SC Integrator using Capacitive T-cell Network

In [24], the capacitors used in the π switched networks were replaced with a capacitive T-cell, as shown in Fig. 3.2. This network reduces the amount of charge delivered to the integrating capacitor (C_2) through the use of larger capacitance values and a capacitive voltage division.

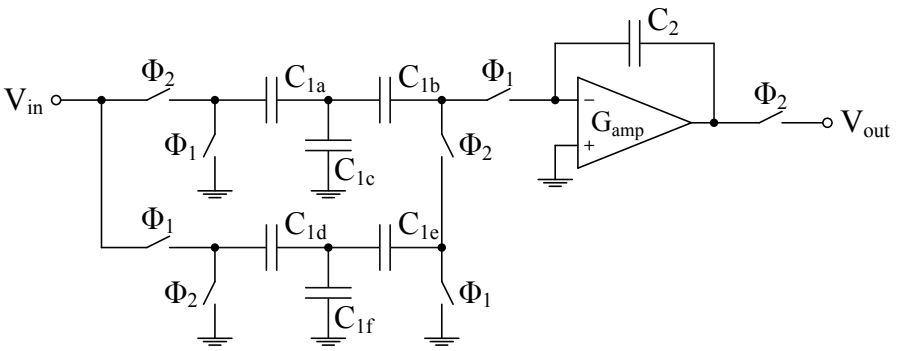


Figure 3.2: SC integrator using capacitive T-cell network.

Considering that the signal is sampled at the end of clock phase Φ_2, the discrete-time transfer function of the SC integrator, using capacitive T-cells, is given by (3.3), where the gain of the amplifier is considered as infinity.

$$H_{T\text{-}Cell}^{\Phi_2}(z) = \frac{C_{1b}C_{1e}(C_{1a}C_{1df} + C_{1d}C_{1ac}\, z^{-1/2})}{C_2\Big(C_{1b}C_{1ac}C_{1df} + C_{1e}\big(C_{1c}C_{1df} + C_{1a}(C_{1b} + C_{1df}) + C_{1b}(C_{1c} + C_{1df})\big)\Big)(z-1)} \tag{3.3}$$

where,

$$C_{1ac} = C_{1a} + C_{1c} \quad \text{and} \quad C_{1df} = C_{1d} + C_{1f} \tag{3.4}$$

Assuming that $C_{1c} = C_{1f} = C_{10}$ and that $C_{1a} = C_{1b} = C_{1d} = C_{1e} = C_{11}$, equation (3.3) can be simplified to:

$$H_{T\text{-}Cell}^{\Phi_2}(z) = \frac{C_{11}^2}{2\, C_2(C_{10} + 2C_{11})}\left(\frac{1 + z^{-1/2}}{z-1}\right) \tag{3.5}$$

In order to obtain the same transfer function as the SC integrator from Section 3.1.1, assuming that both circuits have the same integrating capacitance value (C_2), equation (3.6) has to be satisfied.

$$C_1 = \frac{C_{11}^2}{C_{10} + 2C_{11}} \quad \Rightarrow \quad C_{10} = \frac{C_{11}(-2C_1 + C_{11})}{C_1} \tag{3.6}$$

Table 3.1 shows a design comparison between the basic and the T-cell SC integrators. Using the T-cell network, the capacitance spread was reduced by a factor of 10 and the total capacitance by a factor of 3.4.

Table 3.1: Design comparison between the basic and the T-cell SC integrators.

	Basic SC Integrator		T-cell SC Integrator*		
	$C_{1a,1b}$	C_2	$C_{1a,1b,1d,1e}$	$C_{1c,1f}$	C_2
Capacitance (pF)	0.10	10.00	0.10	0.80	1.00
Capacitance Spread	100.00		10.00		
Total Capacitance (pF)	10.2		3.00		

* Capacitors were scaled so that the smallest capacitors in both SC integrators have the same value.

3.1.3 SC Integrator using Partial Charge Transfer Network

In [25], the switched networks were replaced with a partial charge transfer network, as shown in Fig. 3.3. During one phase (Φ_2), capacitors ($C_{1x,1y,1z}$) sample the input and during the second phase (Φ_1), only part of the charge is transferred to the rest of the circuit. Using this approach it is possible to reduce the capacitor spread.

Considering the influence of parasitic capacitances and that signal is sampled at the end of clock phase Φ_2, using the principle of charge conservation, the discrete-time transfer function of the SC integrator, using partial charge transfer, is given by (3.7).

$$H_{PCT}^{\Phi_2}(z) = \frac{C_{1x}(C_{1z} + C_{p1y} + C_{p1z}) - C_{1y}C_{p1x} + C_{1z}(C_{1x} + C_{p1x})\, z^{-1/2}}{C_2(C_{1x} + C_{1y} + C_{1z} + C_{p1x} + C_{p1y} + C_{p1z})(z-1)} \tag{3.7}$$

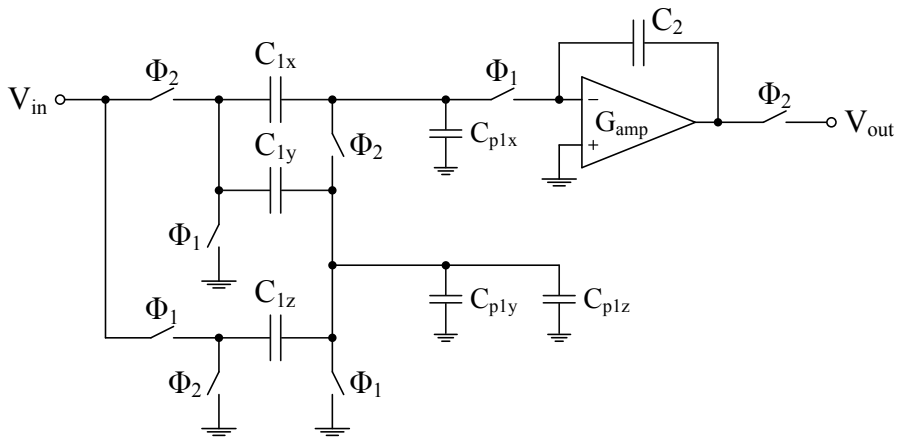


Figure 3.3: SC integrator using partial charge transfer network.

The influence of the parasitic capacitances can be removed from the circuit's transfer function if equation (3.8) is satisfied. Note that, using the same principle, the basic integrator from Section 3.1.1 can also become insensitive to the effect of the parasitic capacitances.

$$\begin{gathered}\delta = \frac{C_{p1x}}{C_{1x}} = \frac{C_{p1y}}{C_{1y}} = \frac{C_{p1z}}{C_{1z}} \Rightarrow \\ H_{PCT}^{\Phi_2}(z) = \frac{C_{1x}C_{1z}(1+\delta)(1+z^{-1/2})}{C_2(C_{1x}+C_{1y}+C_{1z})(1+\delta)(z-1)} = \frac{C_{1x}C_{1z}(1+z^{-1/2})}{C_2(C_{1x}+C_{1y}+C_{1z})(z-1)}\end{gathered} \tag{3.8}$$

In order to obtain the same transfer function as the SC integrator from Section 3.1.1, assuming that both circuits have the same integrating capacitance value (C_2), equation (3.9) has to be satisfied.

$$\frac{C_1}{2} = \frac{C_{1x}C_{1z}}{C_{1x}+C_{1y}+C_{1z}} \quad \Rightarrow \quad C_{1y} = C_{1x}\left(2\,\frac{C_{1z}}{C_1} - 1\right) - C_{1z} \tag{3.9}$$

Table 3.2 shows a design comparison between the basic and the partial charge transfer SC integrators. Using the partial charge transfer topology, the capacitance spread was reduced by a factor of 7.5 and the total capacitance by a factor of 3.6.

Table 3.2: Design comparison between the basic and the partial charge transfer SC integrators.

	Basic SC Integrator		Partial charge transfer SC Integrator*		
	$C_{1a,1b}$	C_2	$C_{1x,1z}$	C_{1y}	C_2
Capacitance (pF)	0.10	10.00	0.10	1.30	1.33
Capacitance Spread	100.00		13.33		
Total Capacitance (pF)	10.2		2.83		

* Capacitors were scaled so that the smallest capacitors in both SC integrators have the same value.

3.1.4 SC Integrator using Split Integrating Capacitor Technique

In the previous examples, the capacitance spread was reduced by using different SC networks at the input, allowing larger sized capacitors to deliver a small amount of charge to the feedback capacitor.

In [26], a similar concept was used in the feedback capacitor, as shown in Fig. 3.4, where a large portion of the input charge is discarded during phase Φ_2, via capacitor C_{2c}, and only a small amount of the input charge is integrated, making the feedback capacitor look larger than it actually is.

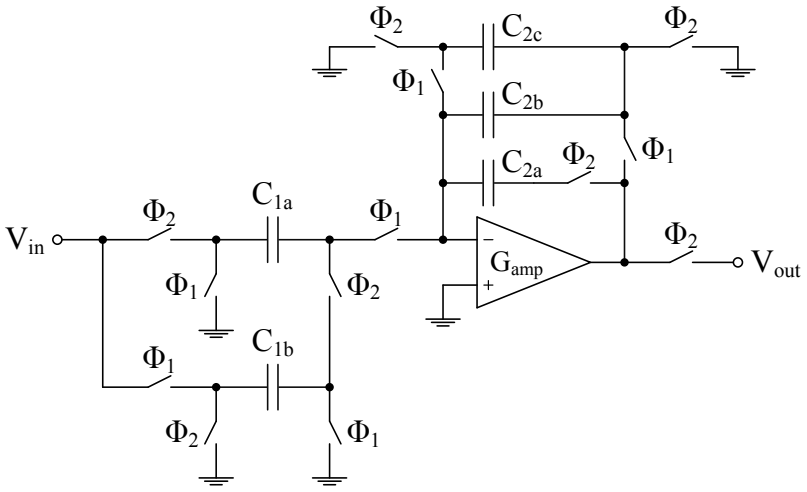


Figure 3.4: SC integrator using split integrating capacitor technique.

Considering that the signal is sampled at the end of clock phase Φ_2, the discrete-time transfer function of the SC integrator, using the split integrating capacitor technique, is given by (3.10), where the gain of the amplifier is considered as infinity.

$$H_{SIC}^{\Phi_2}(z) = \frac{C_{1a}C_{1b}C_{2b}}{(C_{1a}+C_{1b})C_{2a}(C_{2b}+C_{2c})}\left(\frac{1+z^{-1/2}}{z-1}\right) \tag{3.10}$$

In order to obtain the same transfer function as the SC integrator from Section 3.1.1, assuming that both circuits have the same input capacitance value ($C_{1a} = C_{1b} = C_1$), equation (3.11) has to be satisfied.

$$\frac{1}{2\,C_2} = \frac{C_{2b}}{2\,C_{2a}(C_{2b}+C_{2c})} \quad \Rightarrow \quad C_{2a} = \frac{C_2 C_{2b}}{C_{2b}+C_{2c}} \tag{3.11}$$

Table 3.3 shows a design comparison between the basic and the split integrating capacitor SC integrators. Using the split integrating capacitor topology, the capacitance spread was reduced by a factor of 10 and the total capacitance by a factor of 4.6.

Table 3.3: Design comparison between the basic and the split integrating capacitor SC integrators.

	Basic SC Integrator		Split integrating capacitor SC Integrator*			
	$C_{1a,1b}$	C_2	$C_{1a,1b}$	C_{2a}	C_{2b}	C_{2c}
Capacitance (pF)	0.10	10.00	0.10	1.00	0.10	0.90
Capacitance Spread	100.00		10.00			
Total Capacitance (pF)	10.2		2.20			

* Capacitors were scaled so that the smallest capacitors in both SC integrators have the same value.

3.1.5 SC Integrator using Charge Recombination Technique

In [27], a charge recombination technique is presented, using a switched network with the dual function of attenuating the input sample and increase integration, as shown in Fig. 3.5.

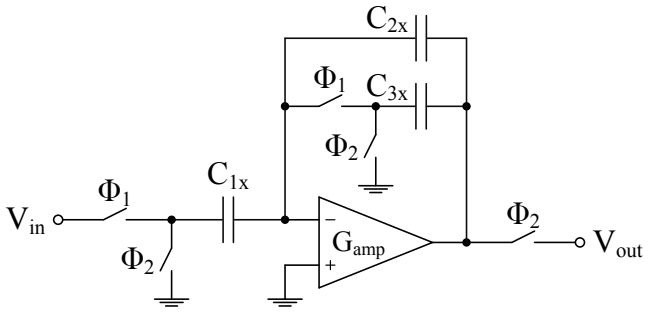


Figure 3.5: SC integrator using charge recombination technique.

Considering that the signal is sampled at the end of clock phase Φ_2, the discrete-time transfer function of the SC integrator, using the charge recombination technique, is given by (3.12), where the gain of the amplifier is considered as infinity.

$$H_{CR}^{\Phi_2}(z) = \frac{C_{1x}C_{3x}}{C_{2x}(C_{2x}+C_{3x})}\left(\frac{z^{-1/2}}{z-1}\right) \tag{3.12}$$

Using this approach, the circuit is insensitive to the effects of the parasitic capacitances, since they are driven by the circuit's input/output or because they are connected to the ground or to virtual ground. However, to get the desired effect, capacitor C_1 needs to be connected to the virtual ground node in both phases. To obtain the same frequency response, assuming the same input capacitance value, equation (3.13) has to be satisfied.

$$\frac{1}{C_2} = \frac{C_{3x}}{C_{2x}(C_{2x}+C_{3x})} \quad \Rightarrow \quad C_{3x} = \frac{C_{2x}^2}{C_2 - C_{2x}} \tag{3.13}$$

Table 3.4 shows a design comparison between the basic and the charge recombination SC integrators. Using the charge recombination topology, the capacitance spread was reduced by a factor of 10 and the total capacitance by a factor of 8.4.

Table 3.4: Design comparison between the basic and the charge recombination SC integrators.

	Basic SC Integrator		Charge recombination SC Integrator*		
	$C_{1a,1b}$	C_2	C_{1x}	C_{2x}	C_{3x}
Capacitance (pF)	0.10	10.00	0.10	1.00	0.11
Capacitance Spread	100.00		10.00		
Total Capacitance (pF)	10.2		1.21		

* Capacitors were scaled so that the smallest capacitors in both SC integrators have the same value.

3.2 Active SC Lowpass Filter using Switched-Current-Assisting Technique

The SC integrator presented in [28], [29] uses a switched-current-assisting (SCA) technique to improve two main challenges in the design of active SC filters due to the switching nature of discrete circuits, the trade-off of speed and linearity with the circuit's power consumption. This technique can be added to existing ones, presented in the 1990's, which relax the performance requirements of the amplifiers. The gain regulation in [30], which uses a control loop to accurately set the gain, and the unity-gain buffer SC integrator in [31], that achieves a higher bandwidth for the same power, when compared with implementations using high-gain amplifiers, however, the circuit becomes sensitive to the effects of parasitic capacitances due to the loss of the virtual ground node.

The SCA technique involves the use of an assisting branch connected to the output of the SC integrator. This branch is used to alleviate the slew rate and bandwidth requirements of the main operational transconductance amplifier (OTA). The circuit's configuration and clock scheme are shown in Fig. 3.6.

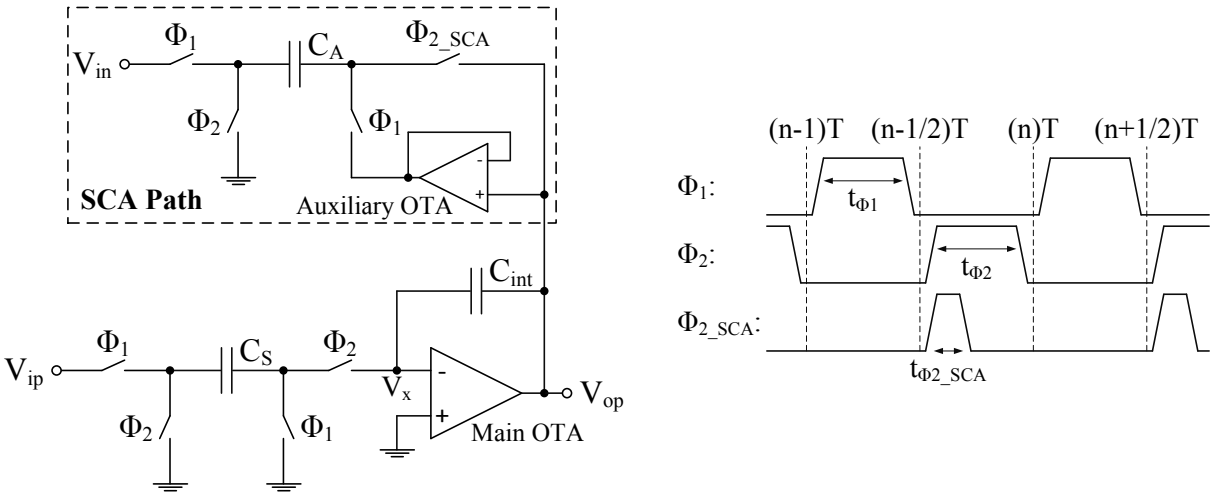


Figure 3.6: Switched-current-assisting SC integrator.

During the sampling phase (Φ_1), the auxiliary OTA, which is in unity-gain configuration, will copy the output voltage into one of the plates of the auxiliary capacitor C_A. During the assisting phase (Φ_{2_SCA}), the amount of charge transferred to the output will be given by (3.14).

$$Q_{C_L}(t_0) = v_{out}^{\Phi_1}\left(1 - e^{-\frac{G_A}{C_A}\,t_{\Phi_1}}\right)\frac{C_A C_L}{C_A + C_L}\left(1 - e^{-\frac{C_A+C_L}{2R_{on}C_A C_L}\,t_{\Phi_{2\_SCA}}}\right) \tag{3.14}$$

where, C_L is the series of C_S with C_{int}, G_A is the transconductance of the auxiliary OTA, R_{on} is the switches resistance in the assisting path, and t_0 is the time at the end of the assisting phase.

Once the assisting phase ends, the SCA branch is disconnected, to avoid its loading effect on the circuit's output, for the remainder of the integration phase (Φ_2).

Without the assisting path, the settling accuracy, which is given by (3.15), is controlled by the transconductance of the main OTA (G_{OTA}), the sampling capacitor C_S, and the length of the integration

phase. Since the phases length is defined by the application and the sampling capacitor affects the gain within the bandwidth, there will be a trade-off in the circuit's settling accuracy between transconductance and power dissipation.

$$v_{out}(t) = -\frac{C_S}{C_{int}}\left(1 - \frac{C_{int}+C_S}{C_S}\, e^{-\frac{G_{OTA}}{C_S}\, t}\right) V_{step} \tag{3.15}$$

With the assisting path, this branch can be designed to move charge at a much faster rate than the main OTA, by controlling the size of C_A and the resistance of the switches (assuming that $\frac{C_A+C_L}{2R_{on}C_AC_L}\, t_{\Phi_{2\_SCA}} \gg \frac{G_A}{C_A}\, t_{\Phi_1}$), accelerating the speed of the integration, i.e., the assisting path will be responsible for moving most of the desired charge to the output during the assisting phase. The remainder of the integration phase will handle the error correction, which is no longer demanding in terms of speed, relaxing the slew rate and bandwidth necessary for the main OTA. Using this technique, a 7/4 improvement was reported in the speed-to-power efficiency of the SC integrator.

The SCA path can also improve the circuit's linearity, by reducing the voltage peak at the virtual ground node during the integration phase. Considering the small signal equivalent circuit shown in Fig. 3.7, without the SCA path, the peak can only be reduced by increasing G_{OTA}, which in turn will increase the power by the same factor (3.16).

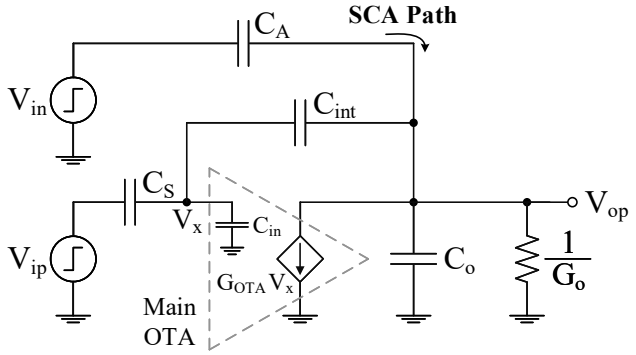


Figure 3.7: Small signal equivalent circuit of the SCA SC integrator.

$$\begin{gathered} V_x(s) = \frac{C_S(C_{int}+C_o)s}{C_{int}G_{OTA} + (C_oC_S + C_{int}C_o + C_{int}C_S)s}, \quad \text{if } C_{in}=0 \text{ and } G_o=0 \Leftrightarrow \\ V_x(t) = \frac{C_S(C_{int}+C_o)}{C_oC_S + C_{int}C_o + C_{int}C_S} e^{-\frac{C_{int}G_{OTA}}{C_oC_S+C_{int}C_o+C_{int}C_S}\, t} \end{gathered} \tag{3.16}$$

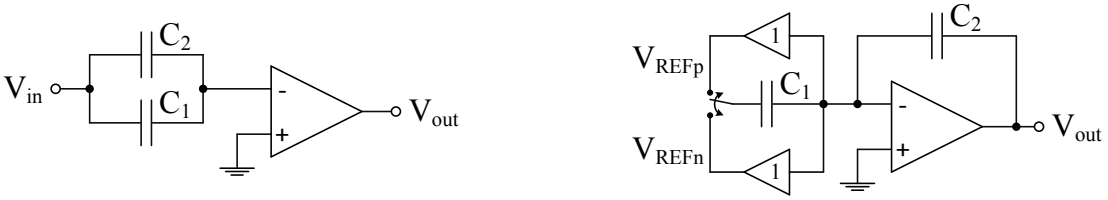
With the assisting path, the SCA network will cause a negative swing at V_x which can, ideally, be matched with the voltage induced by the SC integrator during the integration phase, by carefully sizing capacitor C_A (3.17). In reality, however, the switches resistance and the parasitic capacitances will degrade the efficiency of the voltage cancellation.

$$\begin{gathered} V_x(s) = \frac{(C_S(C_A+C_{int}+C_o) - C_aC_{int})s}{C_S(C_A+C_o)s + C_{int}(G_{OTA} + (C_A+C_o+C_S)s)}, \quad \text{if } C_{in}=0 \text{ and } G_o=0 \\ V_x(s)=0 \Rightarrow C_A = \frac{(C_{int}+C_o)C_S}{C_{int}-C_S} \end{gathered} \tag{3.17}$$

In terms of noise, the S/H noise is not influenced by the SCA path since it only conducts for a short period of time, leaving enough time for the noise induced by the SCA path to be absorbed by the main OTA. During the assisting phase, the auxiliary capacitor is connected to the circuit's output reducing the direct noise generated by the switches and the OTA.

3.3 Virtual Ground Reference Buffer Technique in SC Circuits

In [32], a virtual ground reference buffer (VGRB) technique is described, using as an example a 1-bit flip-around multiplying digital-to-analog converter (MDAC). The technique improves the amplifier feedback factor without altering the signal's gain, breaking away from the conventional inverse relation between the signal gain and the feedback factor, ultimately relaxing the amplifiers bandwidth and noise requirements. The improvement in the circuit's feedback factor is achieved using two level-shifting buffers in parallel with the sampling capacitor, during the charge transfer phase, as shown in Fig. 3.8.



(a) (b)

Figure 3.8: MDAC using VGRB technique during: (a) sampling phase and (b) transfer phase.

In this technique, the reference voltages are referenced to the virtual ground node instead of the system's ground and are generated using the buffers common-mode variation between its input and output. Since the level-shifting buffers replace the existing reference buffers from the conventional architecture, the power penalty is negligible. Table 3.5 shows the comparison of several metrics (neglecting parasitic capacitances) between the conventional MDAC and the VGRB technique, where G_s is the signal gain, f_u is the unity gain bandwidth, and A is the amplifier's gain.

Table 3.5: MDAC performance comparison between conventional approach and VGRB technique.

	Signal gain	Feedback factor	Closed-loop bandwidth	Input-referred noise density⋆	Charge-transfer error
Conventional	$1+\frac{C_1}{C_2}$	$\frac{C_2}{C_1+C_2}$	$\frac{f_u}{G_s}$	$S_{n,oa}(f)$	$\frac{G_s}{A}$
VGRB	$1+\frac{C_1}{C_2}$	1	f_u	$\frac{S_{n,oa}(f)}{G_s^2}$	$\frac{1}{A}$

⋆ Amplifier's input-referred spectral noise density in the respective bandwidths.

Although from an ideal standpoint the influence of capacitor C_1 can be completely removed from the circuit's feedback factor, considering the effect of the parasitic capacitances and the buffers gain, the actual feedback factor is given by $C_2/((1-A_{buffer})C_1+C_2+C_p)$, where A_{buffer} represents the buffers gain.

As described in [33], this technique can be used in different SC circuits, such as SC filters implemented using high-gain amplifiers, relaxing the bandwidth requirements of the amplifiers.

3.4 Recycling SC Buffer Biquad

The motivation behind using filter structures based on unity-gain buffers is due to the minimization of the parasitic capacitances being driven by the amplifier, making it easier to achieve high-frequency bandwidths for the same power, when compared with structures based on high-gain amplifiers [34].

In [2], a recycling SC-buffer biquad is presented, consisting of passive SC networks and an open-loop unity-gain buffer, which behave better to technology downscaling when compared to biquads using high-gain amplifiers and closed-loop negative feedback. Since there is no internal gain, non-linearity and parasitic capacitances are intrinsically low.

The SC buffer shown in Fig. 3.9 can be recycled by sub-dividing the clock period, increasing the number of poles and, saving area and power at the cost of reducing the length of each clock phase.

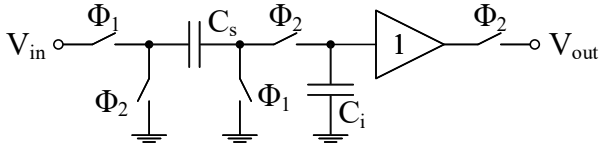


Figure 3.9: First-order SC buffer lowpass filter.

In this configuration, considering the signal is sampled at the end of clock phase Φ_2, using the principle of charge conservation, the SC buffer's discrete-time transfer function is given by (3.18).

$$H^{\Phi_2}(z) = \mathcal{Z}\left(\frac{V_{out}[n]}{V_{in}[n]}\right) = \frac{C_s}{C_i - (C_i + C_s)z} \tag{3.18}$$

If the circuit is recycled once, as shown in Fig. 3.10, a second pole is obtained without the use of an additional buffer, although the existing buffer will need to be 3/2 times faster to compensate the reduction of the duty cycle from 50 % to 33.3 %, while maintaining the same settling accuracy.

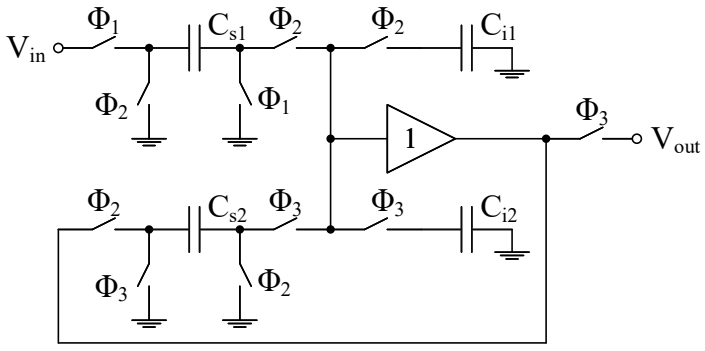


Figure 3.10: SC buffer biquad lowpass filter by recycling the buffer.

Considering the signal is sampled at the end of clock phase Φ_3, using the principle of charge conservation, the recycled SC buffer's discrete-time transfer function is given by (3.19).

$$H^{\Phi_3}(z) = \mathcal{Z}\left(\frac{V_{out}[n]}{V_{in}[n]}\right) = \frac{C_{s1}C_{s2}}{(C_{i1}(z-1)+C_{s1}z)\,(C_{i2}(z-1)+C_{s2}z)} \tag{3.19}$$

This topology, however, can only offer real poles so the roll-off is much lower than a typical Butterworth lowpass filter. To obtain complex poles, two first-order sections can be cascaded with a capacitive feedback (C_i) from the output of the filter to the input of the first buffer, resulting in a simpler structure, when compared with conventional SC integrators [31]. By isolating the passive SC networks with buffers, the capacitor spread is reduced when compared with the Sallen-Key SC biquad [35], improving area and capacitor matching accuracy. The recycled version of this filter is shown in Fig. 3.11.

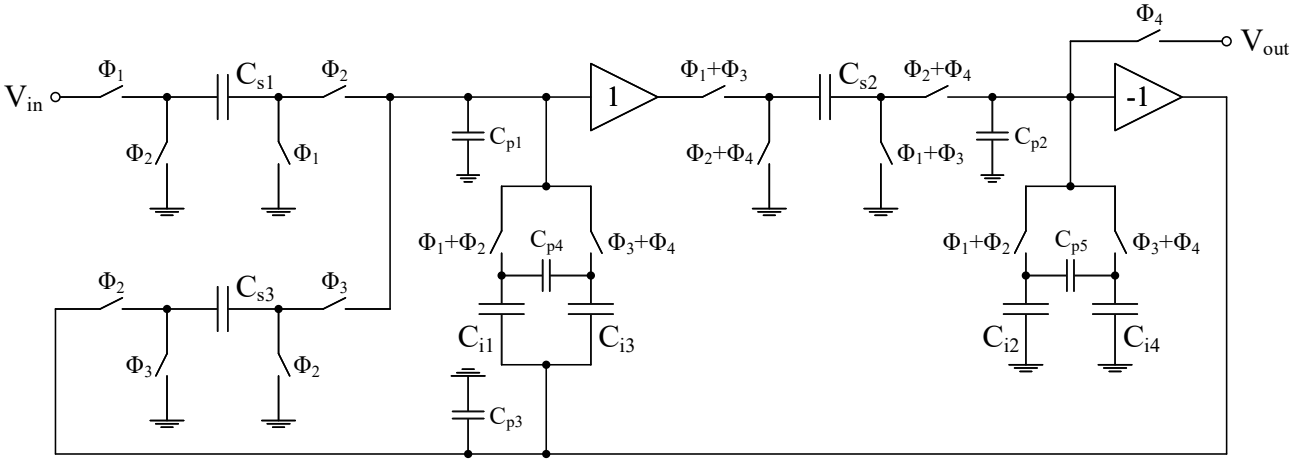


Figure 3.11: Recycling SC-buffer biquad as a 4th-order SC lowpass filter.

By using different sized capacitors for the normal path (phases Φ_1 and Φ_2) and recycled path (phases Φ_3 and Φ_4) the filter can have a different bandwidth and quality factor for each path.

Since the buffer is being recycled, the effect of the parasitic capacitances needs to be carefully considered due to their memory effect in each clock phase, inducing crosstalk. Capacitances C_{p1} and C_{p2} decrease the gain within the passband, capacitances C_{p4} and C_{p5} cause a gain peak near the cutoff frequency, which can be reduced by increasing C_{i1-i4} and, the effect of capacitance C_{p3} can be neglected since is driven by the output.

CHAPTER 4

COMPUTER METHODS FOR THE ANALYSIS AND OPTIMIZATION OF SWITCHED-CAPACITOR CIRCUITS

Over the years, several methods have been developed for the analysis of periodically switched linear networks. One of the earlier works involved the analysis of these networks using a state-space methodology based on unit cisoidal step responses and unit relaxation functions, which are defined by the different switch states [36]. In the 1970's, additional work was presented, using different approaches of the same methodology, more suitable for digital computation [37], [38]. In [39], the state-space methodology was adapted to be used specifically in switched-capacitor (SC) circuits, which, due to the reduced number of different elements, led to significant simplifications. In [40], a more computationally efficient state-space method was presented, where the analysis is valid for any (ideal) switch configuration.

A method to model the settling-time of SC circuits was described in [41]. Although it is quite accurate, the method is presented for SC circuits implemented using the two-stage Miller-compensated amplifier and, even though the model considers the parasitic capacitances from the switches and from the capacitors, it assumes that the RC time constants due to the switches ON resistance, in each node of the circuit, are negligible for the settling performance of the SC circuit. This method was used in an optimization environment, using as design objectives: the equivalent input noise; the output voltage swing; the DC gain of the operational transconductance amplifier (OTA); and the effective number of bits (ENOB) of the complete SC circuit.

In [42], [43], high-level models of complete SC integrator circuits, implemented in Simulink, are used to implement time-domain behavioral models in sigma-delta modulators, improving the computational efficiency over other methods. This approach uses high-level parameters such as the amplifier's DC gain and bandwidth in order to obtain a high-level model of an integrator, thus relying in the need of predetermining the circuit's discrete-time transfer function, before obtaining the model.

Commonly used methods for the analysis of switched networks include: circuit analysis using the principle of charge conservation [44]; the use of the indefinite admittance matrix [45]; 2-port analysis using 4-port equivalent circuits [46]; circuit analysis using Kirchhoff's current law (KCL) - nodal analysis [47]; and modified nodal analysis (MNA) [48]. The MNA is obtained by simplifying the sparse tableau method [49] and determines the circuit's nodal voltages and some branch currents. In [50] the MNA approach was adapted for time, frequency and $\mathcal{Z}$-domain analysis of multi-phase SC networks. These methods are based on the matrix formulation of SC networks, however, topological methods [51] can also be employed, evaluating the graphical representation of the SC network, using signal flow graph (SFG) [52], to obtain the circuit's symbolic transfer function, allowing a better understanding of the discrete-time operation as function of the circuit's components.

A state-of-the-art review of analysis methods for SC circuits, suitable for computer implementation, is presented in [53]. Several tools have been developed using the previously mentioned analysis methods where, some of these tools, have been integrated into optimization environments to find the optimal values of the components, in order to maximize the circuits' performance. Some of these tools are briefly described next.

- **SCANAL** [54] - Simulation software for the analysis of 2-phase SC networks with ideal switches, linear capacitors, ideal opamps and any type of controlled source with constant gain. The software allows sensitivity and frequency response analysis, based on the link-2-port presented in [46] and the MNA [48] approach, using the discrete Fourier transform.
- **DIANA-SC** [55] - Simulation software for the analysis of N-phase SC networks. The software allows time-domain, frequency-domain, sensitivity, and noise analyses for all types of input signals. The analysis method [56] used allows: ideal or resistive switches; parasitic capacitances; finite opamp gain and bandwidth; and non-linearities. The software uses a top-down approach, initially looking at the circuit from an ideal standpoint (ideal zero resistance switches and ideal opamps), performing one circuit evaluation for each clock phase. Afterwards, more accurate models are used in the switches and opamps. In this level, the circuit is evaluated more than once for each clock phase, using backward Euler or trapezoidal integration rules and interpolation techniques. The frequency-domain analysis is performed using the $\mathcal{Z}$-domain transfer matrix [50]. The sensitivity and noise analyses are performed using the adjoint network approach [57].
- **SWITCAP** [58], [59] - Simulation software for the analysis of linear SC and mixed SC/digital networks. The software allows an arbitrary number of phases of unequal duration during a clock period. The SC circuits can include capacitors, voltage-controlled voltage source (VCVS), independent voltage sources and ideal switches controlled by periodic clock signal. Amplifiers with finite bandwidth can be obtained using a combination of ideal elements and resistors can be obtained using equivalent switched networks. Both time-domain and frequency-domain analyses can be performed for different types of input signals (pulse, cosine, exponential, DC), using the nodal analysis method from [60] and the MNA methods from [61], [62].

- **SCAPN** [63] - Simulation software for N-phase SC circuits containing capacitors, switches, VCVS, independent voltage sources, and piecewise-constant input signals. The software performs frequency, sensitivity, and non-linear distortion analyses. The frequency-domain analysis is based on the MNA [50], solved using a pre-LU-factorization algorithm [64] to minimize the computations required to find the solution. The sensitivity analysis, implemented using the adjoint network approach, can be performed on capacitors, capacitor ratios and gain of the controlled sources. The harmonic and intermodulation distortion analyses, implemented using the Volterra series [65] method, evaluates the non-linearity of MOS capacitors, parasitic capacitances, and non-linear gains of the controlled sources using approximated models, consisting of their linear model and charge/voltage distortion sources.
- **SCNSOP** [66] - Simulation and optimization software for N-phase SC circuits. The software allows time-domain, frequency-domain and sensitivity analyses, and poles calculation, without any topology or duty-cycle constraints. The time-domain analysis algorithm is based on the MNA [50]. The frequency domain and sensitivity analyses are based on the adjoint network approach [57]. The software allows the use of ideal capacitors, switches, opamps, and any type of controlled source. A multiple objective (weighted sum) optimization methodology is used to find the required frequency response while minimizing the total capacitance value and the sensitivity of the SC circuit to capacitance, capacitance ratios, and opamp gain variations.
- **WATSNAP** [67] - Periodically switched linear networks simulation and optimization software for continuous and sampled-and-held input signals. The time-domain analysis methodology is based on the two-graph MNA [68]. The frequency-domain and sensitivity analyses are based on the state-space analysis of switched networks [38]. The software allows the use of passive linear elements (resistors, capacitors, and inductors), periodically operated switches, independent and controlled sources, ideal opamps, and non-ideal opamps with finite gain-bandwidth product. WATSNAP uses a gradient algorithm for optimization, in which the goal is to minimize the equiripple in the passband and maximize the stopband attenuation.
- **Spectre** [69] - Simulation software for analog, mixed-signal, and radio frequency circuits. This software is integrated with the Cadence Virtuoso custom design platform and is capable of simulating SC circuits considering non-ideal effects, such as finite bandwidth, device non-linearities, back-gate bias, and slew-rate effects. Spectre provides detailed analyses in multiple domains, including periodic steady state (PSS) and periodic alternating current (PAC) analyses that allow the frequency-domain behavior of SC circuits, at transistor-level, to be modeled with a high degree of accuracy and much quicker when compared with transient simulations of the impulse response.

The previous analysis methods are based on using a simplified model of the SC circuit that can limit the accuracy of the calculation of the frequency response. In some cases this simplified model assumes that the RC time constants of the switches are much lower than the duration of the clock

phases, or in the case where this effect is considered (SWITCAP), it is modeled using an equivalent SC branch. The other limitation of the simplified models is that the amplifiers are replaced by a high-level first-order model, characterized only by the amplifier's DC gain and GBW. This means that all the previous methods are based on an hierarchical approach where the amplifier circuits have to be analyzed separately from the SC filter. All these limitations can, in certain conditions, introduce important inaccuracies in the calculation of the filter's frequency response.

An analysis method to obtain the transfer function of SC circuits, from an ideal standpoint, and two methods to minimize the total capacitance value and the capacitance spread are described in more detail in the following sections.

4.1 Nodal Analysis Method to Obtain the Transfer Function of SC Circuits

In [60], a general method for the analysis of networks containing capacitors, switches and, dependent and independent voltage sources is presented. When compared with other approaches, this method requires a single network topology (capacitance matrix) and one matrix (switching matrix) per switching time, instead of as many networks as there are switching times. By doing this, the circuit topology can be easily analyzed for different switching positions, since only the switching matrices, which are formed of 1's and 0's, need to be changed.

As it was presented, the method does not account for the non-ideal effects of the components, i.e., the switches ON resistance and the amplifiers (dependent voltage sources) slew rate, closed-loop bandwidth, etc.

Using a specific set of matrices, the time-domain voltages in each node, for a particular switch combination, can be calculated using the following system of equations:

$$\begin{array}{c|c|ccc} & & & & \\ p & -E_I^T & \multicolumn{2}{c}{0} & \\ r & D + E_D^T & \multicolumn{2}{c}{} & \\ n & S_k\,C + S_k^T - I & S_k\,E_I & S_k\,E_D & \\ & n & p & r & \end{array} \begin{array}{|c|} v(t) \\ \hline \hat{q}_I(t) \\ \hline \hat{q}_D(t) \\ 1 \end{array} = \begin{array}{|c|c} v_I(t) & p \\ \hline 0 & r \\ \hline S_k\,C\,v(t_k^-) & n \\ 1 & \end{array} \tag{4.1}$$

where, in this system:

- E_I is a $n \times p$ matrix which represents the independent voltage sources, where n is the number of nodes in the circuit (minus the reference node) and p is the number of independent voltage sources.
- E_D is a $n \times r$ matrix representing the dependent voltage sources, where r is the number of dependent voltage sources.
- D is a $r \times n$ matrix which holds the gain coefficients of the dependent voltage sources.

- C is a $n \times n$ matrix, defined as the capacitance matrix where, for $i = j$, C_{ij} holds the total capacitance connected to node i and, for $i \neq j$, the negative of the total capacitance connected between nodes i and j.

- S_k is a $n \times n$ matrix, defined as the switching matrix where, k is the k^{th} switching interval and where, initially, the diagonal of S_k if filled with 1's and with 0's elsewhere. Afterward, if two nodes are connected together through a switch, the '1' representing the highest numbered node is moved to the line of the lowest numbered node, while staying in the same column.

- I is the $n \times n$ identity matrix.

In the particular case of grounded switches, one 0 V dummy independent voltage source needs to be added to the circuit to connect all of these switches together, introducing an additional node to the circuit. This node, however, does not need to be considered in the circuit analysis since its voltage is already know, i.e., 0 V. When one of these switches, which is connected between node n and the dummy voltage source, the n^{th} row and column of the switching matrix S_k is filled with 0's.

The system of equations in (4.1) can be rewritten as a function of the independent input voltages $v_I(t)$ and the initial conditions $v(t_k^-)$:

$$v(t) = A_k\, v_I(t) + B_k\, v(t_k^-) \tag{4.2}$$

where,

$$A_k = (\Phi_k^{-1})_L \qquad \text{and} \qquad B_k = (\Phi_k^{-1})_R\, S_k\, C \tag{4.3}$$

and, considering Φ_k as the matrix that multiplies the unknown vector in (4.1). $(\Phi_k^{-1})_L$ and $(\Phi_k^{-1})_R$ represent, respectively, the upper left $n \times p$ and the upper right $n \times p$ submatrices of Φ_k^{-1}.

Using (4.2), it is possible to determine the voltage at the end of a specific phase, based on the initial conditions, i.e., based on the final voltages from the previous phase.

Considering piecewise-constant inputs, matrices A_k and B_k can be used to obtain the discrete time transfer function $H_l(z)$.

$$H_l(z) = (I - z^{-1}F_l)^{-1}\,(G_l + z^{-1}R_l) \tag{4.4}$$

where,

$$G_l = \sum_{i=1}^{l}\left[\left(\prod_{j=0}^{i-1}\hat{\hat{B}}_{l,j}\right)\hat{\hat{A}}_{l,i}\right], \qquad R_l = \sum_{i=l}^{K}\left[\left(\prod_{j=1}^{i}\hat{\hat{B}}_{l,j}\right)\hat{\hat{A}}_{l,i+1}\right] \qquad \text{and} \qquad F_l = \prod_{i=1}^{K}\hat{\hat{B}}_{l,i} \tag{4.5}$$

and,

$$\hat{\hat{A}}_{l,i} = \begin{cases} A_{l+1-i}, & 1 \leq i \leq l \\ A_{K+l+1-i}, & l \leq i \leq K \\ 0, & i = K+1 \end{cases} \qquad \text{and} \qquad \hat{\hat{B}}_{l,i} = \begin{cases} I, & i = 0 \\ B_{l+1-i}, & 1 \leq i \leq l \\ B_{K+l+1-i}, & l \leq i \leq K \end{cases} \tag{4.6}$$

with K equal to the number of switch combinations and $1 \leq l \leq K$.

Although this method does not consider the non-ideal effects of the switches (ON resistance) and amplifiers (closed-loop bandwidth, slew-rate, etc), it can be efficiently implemented in an optimization environment since matrices D, E_D, E_I, C and S_K can be directly obtained from inspecting the circuit and most of these matrices positions are filled with 0's, making computations faster.

In this method there is a dedicated matrix for the switches positions, making it well suited to quickly evaluate circuit topologies, from an ideal standpoint, and to test switches combinations, since only the switching matrices need to be changed.

4.1.1 Example: Second-Order Lowpass SC Filter

A second-order lowpass SC filter, which is show in Fig. 4.1, is used to exemplify the nodal analysis method described in Section 4.1.

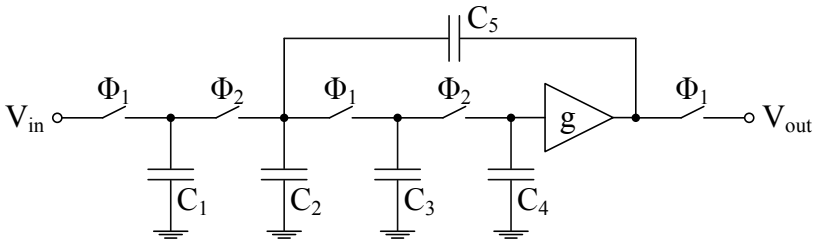


Figure 4.1: Second-order lowpass SC filter

Fig. 4.2 shows a simplified version of the filter, more suitable for the extraction of the necessary matrices.

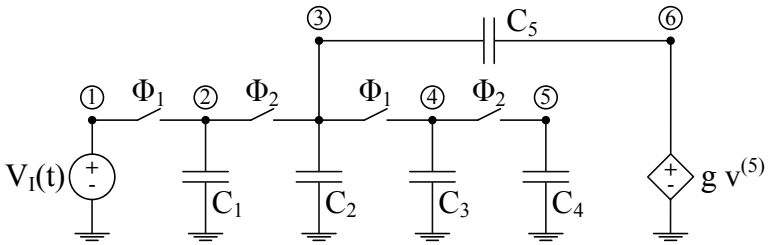


Figure 4.2: Simplified second-order lowpass SC filter

The matrices which describe the independent and dependent voltage sources in the filter are shown in (4.7).

$$
\begin{aligned}
D &= \begin{bmatrix} 0 & 0 & 0 & 0 & g & 0 \end{bmatrix} \\
E_I &= \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
E_D &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}^T
\end{aligned}
\tag{4.7}
$$

The capacitor matrix C, which is valid for every switch combination, and the switching matrices S_k are shown in (4.8).

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_2+C_5 & 0 & 0 & -C_5 \\ 0 & 0 & 0 & C_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_4 & 0 \\ 0 & 0 & -C_5 & 0 & 0 & C_5 \end{bmatrix}, \quad S_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad S_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.8}$$

where, S_1 and S_2 represent the switching matrices during phase Φ_1 and phase Φ_2, respectively.

Using the vectors in (4.7) and matrices C and S_1 from (4.8), matrix Φ_1 is obtained, which can then be used to calculate matrices A_1 and B_1.

$$A_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad B_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha & \beta & g\,\gamma & -\gamma \\ 0 & 0 & \alpha & \beta & g\,\gamma & -\gamma \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & g & 0 \end{bmatrix} \tag{4.9}$$

where,

$$\alpha = \frac{C_2+C_5}{C_2+C_3+C_5}, \quad \beta = \frac{C_3}{C_2+C_3+C_5}, \quad \gamma = \frac{C_5}{C_2+C_3+C_5} \tag{4.10}$$

Solving the system of equations in (4.2), the nodal voltages at the end of any phase Φ_1 can be obtained based on (4.11) and the initial conditions, i.e., the end voltages of the previous phase (Φ_2).

$$\begin{bmatrix} v_1^{\Phi_1}(t) \\ v_2^{\Phi_1}(t) \\ v_3^{\Phi_1}(t) \\ v_4^{\Phi_1}(t) \\ v_5^{\Phi_1}(t) \\ v_6^{\Phi_1}(t) \end{bmatrix} = \begin{bmatrix} v_I(t) \\ v_I(t) \\ \alpha\, v_3(t_k^-) + \beta\, v_4(t_k^-) + g\,\gamma\, v_5(t_k^-) - \gamma\, v_6(t_k^-) \\ \alpha\, v_3(t_k^-) + \beta\, v_4(t_k^-) + g\,\gamma\, v_5(t_k^-) - \gamma\, v_6(t_k^-) \\ v_5(t_k^-) \\ g\, v_5(t_k^-) \end{bmatrix} \tag{4.11}$$

Matrices A_2 and B_2 are obtained in the same manner as matrices A_1 and B_1 but using the switching matrix for phase Φ_2 (S_2).

$$A_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_1\,\delta & \zeta & g\,C_3\,\lambda & g\,C_4\,\lambda & -C_5\,\delta \\ 0 & C_1\,\delta & \zeta & g\,C_3\,\lambda & g\,C_4\,\lambda & -C_5\,\delta \\ 0 & 0 & 0 & C_3\,\xi & C_4\,\xi & 0 \\ 0 & 0 & 0 & C_3\,\xi & C_4\,\xi & 0 \\ 0 & 0 & 0 & g\,C_3\,\xi & g\,C_4\,\xi & 0 \end{bmatrix} \tag{4.12}$$

where,

$$\delta = \frac{1}{C_1+C_2+C_5}, \quad \zeta = \frac{C_2+C_5}{C_1+C_2+C_5}, \quad \lambda = \frac{C_5}{(C_3+C_4)\,(C_1+C_2+C_5)}, \quad \xi = \frac{1}{C_3+C_4} \tag{4.13}$$

The voltages at the end of any phase Φ_2 are then obtained based on 4.14.

$$\begin{bmatrix} v_1^{\Phi_2}(t) \\ v_2^{\Phi_2}(t) \\ v_3^{\Phi_2}(t) \\ v_4^{\Phi_2}(t) \\ v_5^{\Phi_2}(t) \\ v_6^{\Phi_2}(t) \end{bmatrix} = \begin{bmatrix} v_I(t) \\ C_1\,\delta\,v_2(t_k^-) + \zeta\,v_3(t_k^-) + g\,\lambda\,(C_3\,v_4(t_k^-) + C_4\,v_5(t_k^-)) - C_5\,\delta\,v_6(t_k^-) \\ C_1\,\delta\,v_2(t_k^-) + \zeta\,v_3(t_k^-) + g\,\lambda\,(C_3\,v_4(t_k^-) + C_4\,v_5(t_k^-)) - C_5\,\delta\,v_6(t_k^-) \\ \xi\,(C_3\,v_4(t_k^-) + C_4\,v_5(t_k^-)) \\ \xi\,(C_3\,v_4(t_k^-) + C_4\,v_5(t_k^-)) \\ g\,\xi\,(C_3\,v_4(t_k^-) + C_4\,v_5(t_k^-)) \end{bmatrix} \tag{4.14}$$

The transfer function of the circuit is also calculated based on matrices $A_{1,2}$ and $B_{1,2}$. Matrices G_l, R_l and, F_l are calculated based on (4.5) and (4.6) and, on the phase of interest. Considering that the circuit has two clock phases and that the output is sampled at the end of phase Φ_1, $l = 1$ and $K = 2$.

$$G_1 = A_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad R_1 = B_1\,A_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$F_1 = B_1\,B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_1\,\alpha\,\delta & \alpha\,\zeta & C_3\,(g\,\alpha\,\lambda + \beta\,\xi) & C_4\,(g\,\alpha\,\lambda + \beta\,\xi) & -C_5\,\alpha\,\delta \\ 0 & C_1\,\alpha\,\delta & \alpha\,\zeta & C_3\,(g\,\alpha\,\lambda + \beta\,\xi) & C_4\,(g\,\alpha\,\lambda + \beta\,\xi) & -C_5\,\alpha\,\delta \\ 0 & 0 & 0 & C_3\,\xi & C_4\,\xi & 0 \\ 0 & 0 & 0 & g\,C_3\,\xi & g\,C_4\,\xi & 0 \end{bmatrix} \tag{4.15}$$

Using (4.15) the transfer function of the circuit, during phase Φ_1, is obtained and, is identical to the one obtained from using the charge conservation principle.

$$H^{\Phi_1}(z) = \frac{a_0}{b_0 + b_1\,z + b_2\,z^2} \tag{4.16}$$

where,

$$\begin{aligned} a_0 &= -\,g\,C_1\,C_3\,(C_2 + C_5) \\ b_0 &= -\,(C_2 + C_5)(g\,C_3\,C_5 + C_4\,(C_2 + C_5)) \\ b_1 &= C_1\,C_3\,(C_3 + C_4) + C_2^2\,(C_3 + 2\,C_4) + C_1\,C_4\,C_5 + C_5\,(C_3\,(C_3 + C_4) + (C_3 + g\,C_3 + 2\,C_4)\,C_5) + \\ &\quad + C_2\,(C_3^2 + C_4\,(C_1 + 4\,C_5) + C_3\,(C_4 + (2 + g)\,C_5)) \\ b_2 &= -\,(C_3 + C_4)\,(C_1 + C_2 + C_5)\,(C_2 + C_3 + C_5) \end{aligned} \tag{4.17}$$

4.2 Topology Exploration for Optimal Capacitance Sizing in SC Biquads

The minimization of the total capacitance value in SC biquads has been researched extensively since they can be used as the basic building block for the realization of high-order filters, by cascading several biquad sections [26], [70]. In the literature, only a limited number of (conventional) topologies are typically considered for exploration and are analytical-based, where the decision on which capacitors to set is based on the designer's experience and in heuristic rules which, depending on the biquad's topology and input specifications, may not give the best solution.

In [71], an automated systematic design flow for the optimal capacitance sizing problem, at block-level, is presented. In this work, all possible topologies with two integrators (one pole per ampop) and using three different types of branches are considered. The SFG of a possible biquad template configuration is shown in Fig. 4.3.

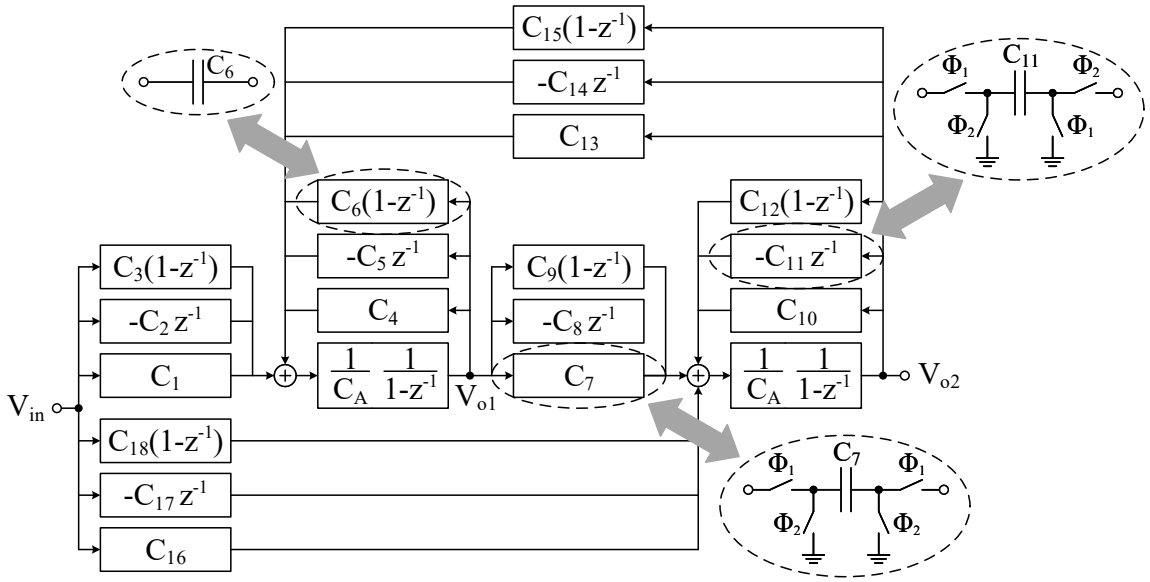


Figure 4.3: Signal flow graph of the biquad template.

This methodology receives, as inputs, the desired DC gain, the cutoff frequency, the quality factor, and the SC filter's sampling frequency. This information is then used to calculate the numerical discrete-time transfer function, either by using digital filter design tools, such as MATLAB, or by converting a second order system's continuous-time transfer function into discrete-time using the bilinear transformation (4.18).

$$H(s) = \frac{\omega_p^2}{\omega_p^2 + \frac{\omega_p}{Q_p}s + s^2} \quad \Leftrightarrow$$
$$\Leftrightarrow H\left(\frac{2}{T_s}\frac{z-1}{z+1}\right) = \frac{\omega_p^2\, T_s^2\, (1+z)^2}{4+\omega_p^2\, T_s^2\, -2\,\frac{\omega_p}{Q_p}T_s + \left(2\,\omega_p^2\, T_s^2 - 8\right) z + \left(\omega_p^2\, T_s^2 + 2\,\frac{\omega_p}{Q_p}T_s + 4\right) z^2} \tag{4.18}$$

where, ω_p is the cutoff frequency, Q_p is the poles' quality factor, and T_s is the filter's sampling period.

The symbolic transfer function can be obtained either by charge conservation equations or through SFG analysis. In addition to the requirement of strictly matching the template's transfer function to the prototype, which is obtained using the desired specifications, other performance metrics can be evaluated, such as the circuit's noise.

The optimal capacitance sizing problem is solved using a mixed integer nonlinear programmming (MINLP) software with a set of constraints: matching of the template and prototype transfer functions; an output swing constraint; and capacitor values within the design space. The software also uses in the cost function: the minimization of the total number of capacitors needed (controlled by binary operators); the total capacitance value; and the poles' quality factor sensitivity. This methodology uses an additional constraint to avoid direct charge coupling between the circuit's input and output by not allowing branches C_1 and C_7 at the same time or C_3 and C_9.

Using this methodology it is possible to automatically optimize the capacitance sizing considering different topologies, using a single biquad template, obtaining the best architecture and design for a given application.

4.3 Optimal Capacitance Sizing of SC Filters using Linear Programming and Simulated Annealing

In [72], a comparative study between a linear programming (LP) algorithm (deterministic) and a simulated annealing (SA) algorithm (stochastic), used for the optimal capacitance sizing of SC filters is presented, using the Fleischer-Laker architecture [73], shown in Fig. 4.4, as an example.

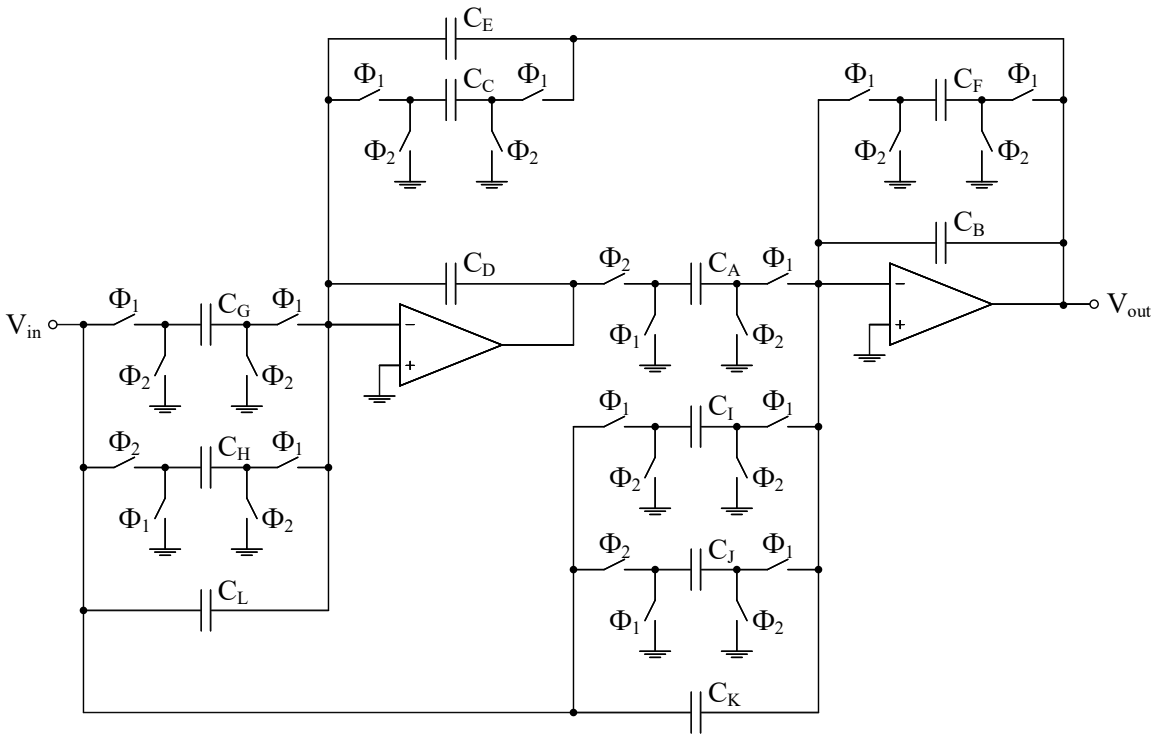


Figure 4.4: Fleischer-Laker SC filter.

The optimization problem is formulated as a linear programming problem using: the sum of all capacitances in the filter, which is the objective function to be minimized; the numerical prototype transfer function ($H_{prototype}$); and the circuit's symbolic transfer function. Using the prototype and symbolic transfer functions, a system of six equations (constraints) is obtained (4.19). However, these equations are nonlinear, resulting in a nonlinear programming problem, where finding the global minimum is difficult since it is not known if the problem is convex.

$$H_{prototype}(z) = -\frac{n_0 + n_1\, z^{-1} + n_2\, z^{-2}}{d_0 + d_1\, z^{-1} + d_2\, z^{-2}} \Rightarrow \begin{cases} n_0 = C_D C_K + C_D C_I \\ n_1 = -2\, C_D C_K - C_D C_J - C_D C_I + C_G C_A + C_L C_A \\ n_2 = C_D C_K + C_D C_J - C_L C_A - C_H C_A \\ d_0 = C_D C_B + C_D C_F \\ d_1 = -2\, C_D C_B - C_D C_F + C_A C_E + C_A C_C \\ d_2 = C_D C_B - C_A C_E \end{cases} \tag{4.19}$$

The LP problem was reformulated into a parametric LP problem, considering linear variations in the values of two capacitors, while the others remain fixed. With this approach, the system of nonlinear equations can be transformed into a linear problem, that is convex, making it possible to find a unique global minimum in the domain of the two variable capacitors. Considering capacitors C_A and C_D as the free parameters, since they appear most often in the equations, the linear system of equations, used as constraints in the LP problem, is given by (4.20).

$$\begin{cases} H_{prototype}(1) = -\dfrac{n_0 + n_1 + n_2}{d_0 + d_1 + d_2} = \dfrac{\overline{n}}{\overline{d}} \\ H_{prototype}(\infty) = \dfrac{n_0}{d_0} \\ H_{prototype}(0) = \dfrac{n_2}{d_2} \end{cases} \Rightarrow \begin{cases} \overline{d} = C_A C_C \\ d_0 = C_D(C_B + C_F) \\ d_2 = C_D C_B - C_A C_E \\ \overline{n} = C_A(C_G - C_H) \\ n_0 = C_D(C_K + C_I) \\ n_2 = C_D(C_K + C_J) - C_A(C_L + C_H) \end{cases} \tag{4.20}$$

It was reported in [72] that, using a 48 x 48 grid (2304 samples) for the values of C_A and C_D, the optimized transfer function, using the parametric LP algorithm, is obtained in approximately 9 s. However, the optimized solutions do not take into consideration the circuit's capacitance spread, and solutions with very large capacitor ratios can be obtained.

Considering the capacitance spread as an additional constraint, it is necessary to use an iterative approach to solve the LP problem, slowly increasing the minimum allowed value of the smallest capacitor until a good compromise between capacitance spread and area is achieved, since convexity is lost when considering the capacitance spread as a constraint.

The SA algorithm was used to determine its capability in finding suboptimal solutions with comparable performance to those obtained from the LP algorithm and to determine the amount of time taken to find those solutions. The SA algorithm introduces 1 % variations in the capacitor values in each iteration. If a smaller area is achieved, the result is updated, otherwise, the result only has a probability $p = e^{\Delta \mathcal{J} / T_k}$ of being updated, where $\Delta \mathcal{J}$ represents the variation in total capacitance between solutions and T_k the temperature of the current iteration, that has a exponential decay with the number of iterations. This approach prevents the algorithm from getting stuck at a local minimum and, given enough time, the algorithm can even find the global minimum.

From the comparison made in [72], it was concluded that both methods produce similar results in terms of total capacitance area and capacitance spread. However, due to the stochastic nature of the SA algorithm, it is necessary to perform multiple runs of this algorithm to find suboptimal solutions with comparable performance to the ones obtained from the LP algorithm, making the SA approach substantially slower, taking 20 s per run.

CHAPTER 5

ANALYSIS METHODOLOGIES FOR THE DESIGN OF SWITCHED-CAPACITOR FILTERS

In the design of circuits or systems, a topology, or set of topologies, are chosen based on accumulated experience. These topologies are then explored to find the one that offers the best performance in terms of power consumption, chip area, sensitivity, complexity, noise, etc.

In the case of switched-capacitor (SC) filters, this means that it is necessary to analyze the circuit in each clock phase to understand its behavior as a function of the components' values and obtain its transfer function to verify if it complies with the desired specifications. Since, in most cases, it is necessary to check the performance of the circuit for different component values (Monte Carlo) and for different process corners, it is important to have a computationally efficient method to quickly determine the transfer function of a SC circuit, considering the non-ideal effects of the components.

In this chapter, two methods for the analysis of SC filters are described. In the first method (symbolic), described in Section 5.1, the SC filter is analyzed hierarchically [4]–[6]. Initially the filter is designed from an ideal standpoint. Afterwards, the remaining blocks in the circuit (amplifier and switches) are designed considering the sizing obtained for the ideal filter. This method is best suited for simple circuits since the design of the amplifier and the switches relies on the closed-loop symbolic equation of the amplifier with the filter, which can result in a high order transfer function even for one stage amplifiers, increasing the computation effort of calculating its inverse Laplace transform to convert the frequency-domain equation into the time-domain. In the second method (numeric), described in Section 5.2, the SC filter is analyzed using a non-hierarchical approach [3], where the non-ideal effects of the transistors (in the amplifier and in the switches) are taken into consideration, allowing the accurate computation of the frequency response, even in the case of incomplete settling in the SC branches. Since this analysis method is purely numeric in nature, it can be efficiently implemented in a computer and be used to analyze more complex circuits without increasing the complexity of the computations.

5.1 Hierarchical Symbolic Methodology for the Analysis of SC Filters

5.1.1 Method Description

The hierarchical symbolic methodology for the analysis of SC circuits is divided into three steps: the first step consists in designing the SC filter using its discrete-time symbolic transfer function, obtained through the use of the charge conservation principle, and considering all the components as ideal; the second step consists in designing the amplifier circuit, at transistor level, in order to have the necessary gain and enough closed-loop bandwidth to replace the ideal gain block used in the previous step; finally, the third step consists in designing the switches to have a RC time constant that is small enough to charge the capacitors in the phases' duration.

Note that, the design of the amplifier and of the switches RC time constants are dependent on each other, i.e., it is necessary to have an estimate of the switches ON resistance for the calculation of the amplifier closed-loop transfer function and an estimate for the amplifier's input and input-to-output parasitic capacitances for the calculation of the switches RC time constants. In this thesis, the design of the amplifier is performed first, using a rough estimate for the value of the switches ON resistance, which is dependent on the length of the clock phases and on the voltage value driving the transistors operating as switches.

5.1.1.1 Design of the Ideal SC Filter

In this step, the SC filter is designed from an ideal standpoint, i.e, the switches are considered as open-circuits when they are open and as short-circuits when they are closed, and the amplifier is considered as a voltage-controlled voltage source (VCVS) with infinite bandwidth. The SC filter is designed using its discrete-time transfer function $H(z)$, which is obtained using the charge conservation principle.

5.1.1.2 Design of the Amplifier

After an ideal design is obtained for the SC filter, the amplifier is designed, at transistor level, to have the necessary gain and enough bandwidth to replace the ideal VCVS used in the previous step. This is done using the time-domain methodology presented in [74], which ensures that, if a given settling error is reached within the desired settling time, then the amplifier has enough open-loop gain, closed-loop bandwidth and slew-rate.

An approximation for the settling time of the amplifier can be calculated using the circuits' closed-loop transfer function, in each clock phase. This approach is based on the feedback theory used in continuous time circuits [75]. The general closed-loop structure is shown in Fig. 5.1.

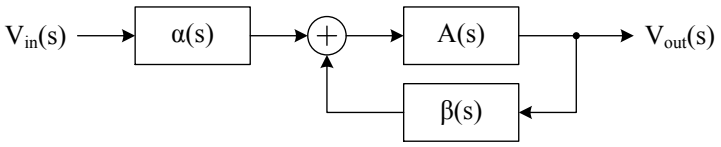


Figure 5.1: General closed-loop signal-flow diagram.

where, $A(s)$ is the amplifier's open-loop gain, $\beta(s)$ is the feedback factor, and $\alpha(s)$ the input factor, all shown in (5.1). When calculating the open-loop gain $A(s)$, it is necessary to consider the filters' load effect (Z_{22}) seen from the amplifier's output. Note that the equations of the feedback factor and input factor are calculated considering the capacitive effects of the amplifier circuit, i.e, input and input-to-output parasitic capacitances.

$$A(s) = \frac{v_{amp_{out}}}{v_{amp_{in}}} \qquad \beta(s) = \left.\frac{v_{amp_{in}}}{v_{amp_{out}}}\right|_{v_{in}=0} \qquad \alpha(s) = \left.\frac{v_{amp_{in}}}{v_{in}}\right|_{v_{out}=0} \tag{5.1}$$

Considering the signal-flow diagram, shown in Fig. 5.1, the closed-loop transfer function is given by (5.2).

$$A_{fb}(s) = \frac{V_{out}(s)}{V_{in}(s)} = \frac{\alpha(s)\ A(s)}{1 - \beta(s)\ A(s)} \tag{5.2}$$

The settling time can then be obtained by using the inverse Laplace transform to equation (5.1) multiplied by $1/s$, which corresponds to applying a unity step at the input of the filter.

$$v_{out}(t) = \mathcal{L}^{-1}\left(\frac{1}{s}\ A_{fb}(s)\right) \tag{5.3}$$

Depending on the complexity of the circuit, it might not be possible to analytically calculate the inverse Laplace transform of the closed-loop transfer function $A_{fb}(s)$. In this case, it is necessary to calculate the partial-fraction decomposition[1] [76] of the closed-loop transfer function to decompose $A_{fb}(s)$ into several, smaller order, transfer functions (typically, first order).

Considering a typical SC circuit, where $num(s)/den(s)$ is a proper rational function, i.e., the order of $den(s)$ is higher than that of $num(s)$, and where there are no repeated roots, the closed-loop transfer function can be written in the form shown in (5.4).

$$\begin{aligned} A_{fb}(s) = \frac{num(s)}{den(s)} &= \frac{a_0\ s^0\ +\ a_1\ s^1\ +\ a_2\ s^2\ +\ \ldots\ +\ a_{n-1}\ s^{n-1}}{b_0\ s^0\ +\ b_1\ s^1\ +\ b_2\ s^2\ +\ \ldots\ +\ b_n\ s^n} \\ &= \frac{c_1}{s-p_1} + \frac{c_2}{s-p_2} + \frac{c_3}{s-p_3} + \ \ldots\ + \frac{c_n}{s-p_n} \end{aligned} \tag{5.4}$$

where, p_n are the roots of the polynomial $den(s)$, i.e., the poles of the closed-loop transfer function and c_n the unknown constants that need to be calculated. Reducing the decomposed fractions to the same denominator and multiplying both sides of the equation by $den(s)$, (5.4) can be rewritten to the form shown in (5.5).

$$\begin{aligned} a_0\ s^0 + a_1\ s^1 + a_2\ s^2 + \ \ldots\ + a_{n-1}\ s^{n-1} = {} & c_1\ (s-p_2)(s-p_3)\ \ldots\ (s-p_{n-2})(s-p_{n-1})(s-p_n) + \\ & +\ c_2\ (s-p_1)(s-p_3)\ \ldots\ (s-p_{n-2})(s-p_{n-1})(s-p_n) + \\ & \qquad\qquad \vdots \\ & +\ c_n\ (s-p_1)(s-p_2)(s-p_3)\ \ldots\ (s-p_{n-2})(s-p_{n-1}) \end{aligned} \tag{5.5}$$

[1] Since in a typical SC circuit it is very rare to have repeated roots in $den(s)$, due to precision errors, only the non-repeatable roots case of partial-fraction decomposition is presented in this thesis.

Since there are n unknown variables but only one equation, it is not possible to directly calculate all the unknowns. However, since the equation is true for any value of s, then it must also be true for the specific values of the roots of $den(s)$ (p_1, p_2, p_3, ..., p_n). In this case, when s is replaced by a value of p_n, all other coefficients will disappear from equation (5.5) and the only unknown remaining will be coefficient c_n. Doing this procedure for all the root values of $den(s)$, it is possible to determine all values of c_n, as shown in (5.6).

$$\begin{aligned} c_1 &= \left.\frac{a_0\, s^0 \;+\; a_1\, s^1 \;+\; a_2\, s^2 \;+\; \ldots \;+\; a_{n-1}\, s^{n-1}}{(s-p_2)\,(s-p_3)\;\ldots\;(s-p_n)}\right|_{s=p_1} \\ c_2 &= \left.\frac{a_0\, s^0 \;+\; a_1\, s^1 \;+\; a_2\, s^2 \;+\; \ldots \;+\; a_{n-1}\, s^{n-1}}{(s-p_1)\,(s-p_3)\;\ldots\;(s-p_n)}\right|_{s=p_2} \\ &\vdots \\ c_n &= \left.\frac{a_0\, s^0 \;+\; a_1\, s^1 \;+\; a_2\, s^2 \;+\; \ldots \;+\; a_{n-1}\, s^{n-1}}{(s-p_1)\,(s-p_2)\;\ldots\;(s-p_n-1)}\right|_{s=p_n} \end{aligned} \tag{5.6}$$

After obtaining the values of c_n from (5.6), the inverse Laplace transform of the decomposed fractions (5.4) can be calculated from (5.7).

$$v_{out}(t) = \sum_{i=1}^{n}\left[\mathcal{L}^{-1}\left(\frac{1}{s}\,\frac{c_i}{s-p_i}\right)\right] = \sum_{i=1}^{n}\left[\frac{c_i}{p_i}\Big(-1+e^{p_i\,t}\Big)\right] \tag{5.7}$$

5.1.1.3 Design of the Switches RC Time Constants

The RC time constants of the switches can be calculated using a similar method to the one used to calculate the amplifier's closed-loop transfer function, however, since they require the computation of the inverse Laplace transform, these methods are computationally intensive. Instead, first-order RC approximations can be used to estimate the switches RC time constants, which give good approximations except for nodes that are strongly dependent on the amplifier's bandwidth.

The first-order RC approximations are calculated using equivalent circuits, where all switches in the active phase are considered as short-circuits (except for the one which time constant is being calculated, which is considered as a resistor) and all switches in the opposite phase as open-circuits. Note that the equivalent circuits only consider the effects of the input and the input-to-output parasitic capacitances from the amplifier circuit and not the complete small signal model of the amplifier.

The time constants of the switches can then be used to calculate the amount of time it takes to reach a given settling error, which can be calculated from (5.8), where τ represents the switches' time constant.

$$Settling\ Error(\%) = 100 \times e^{-t/\tau} \qquad \Leftrightarrow \qquad t = -\tau\ \log\Big[Settling\ Error(\%)/100\Big] \tag{5.8}$$

During this step, it is also necessary to compensate the values of the switches' parasitic capacitances into the main SC filter capacitors.

5.1.2 Example: Second-Order Lowpass SC Filter

In this section, a second-order lowpass SC filter, shown in Fig. 5.2, is used to demonstrate the analysis method presented in Section 5.1.1. The filter architecture is based on the continuous-time lowpass Sallen-Key topology [35], replacing the resistors with parallel SC branches [77] and adding capacitors C_2 and C_5 to allow the compensation of the parasitic capacitances in every node of the filter. This architecture is capable of implementing lowpass filtering functions using low gain amplifiers (voltage-buffers) with gain below unity and the filter's low frequency gain is given by the amplifier's DC gain.

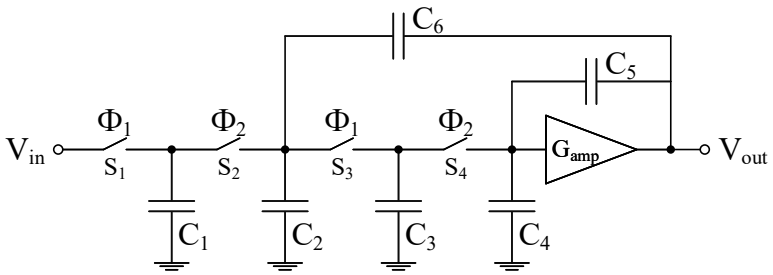


Figure 5.2: Second-order lowpass SC filter.

5.1.2.1 Design of the ideal SC Filter

Considering the equivalent circuits during phase Φ_1 and phase Φ_2, shown in Fig. 5.3, where the amplifier block is considered as an ideal block (VCVS), using the principle of charge conservation and considering that the output is sampled at the end of clock phase Φ_1, the system of charge equations (5.9) is obtained.

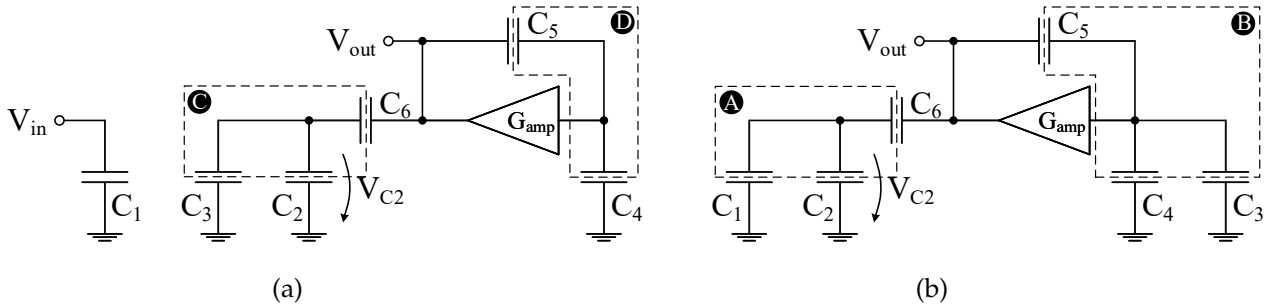


Figure 5.3: Equivalent circuit during (a) phase Φ_1 and (b) phase Φ_2.

$$
\begin{aligned}
&\text{(A)}\; V_{in}[n-1]\,C_1 + V_{C2}[n-1]\,(C_2+C_6) - V_{out}[n-1]\,C_6 = V_{C2}[n-1/2]\,(C_1+C_2+C_6) - V_{out}[n-1/2]\,C_6 \\
&\text{(B)}\; V_{C2}[n-1]\,C_3 + V_{out}[n-1]\left(\frac{C_4+C_5}{G_{amp}} - C_5\right) = V_{out}[n-1/2]\left(\frac{C_3+C_4+C_5}{G_{amp}} - C_5\right) \\
&\text{(C)}\; V_{C2}[n-1/2]\,(C_2+C_6) + V_{out}[n-1/2]\left(\frac{C_3}{G_{amp}} - C_6\right) = V_{C2}[n]\,(C_2+C_3+C_6) - V_{out}[n]\,C_6 \\
&\text{(D)}\; V_{out}[n-1/2]\left(\frac{C_4+C_5}{G_{amp}} - C_5\right) = V_{out}[n]\left(\frac{C_4+C_5}{G_{amp}} - C_5\right)
\end{aligned}
\tag{5.9}
$$

Solving the system of charge equations (5.9) and using the $\mathcal{Z}$-transform, the discrete-time transfer function (5.10) is obtained.

$$H_{filter}^{\Phi_1}(z) = \mathcal{Z}\left(\frac{V_{out}[n]}{V_{in}[n]}\right) = \frac{n_0}{d_0 + d_1\, z + d_2\, z^2} \tag{5.10}$$

where,

$$\begin{aligned}
n_0 =& C_1 C_3\, (C_2 + C_6)\, G_{amp} \\
d_0 =& (C_2 + C_6)\, ((C_4 + C_5)\, (C_2 + C_6) - (C_5 (C_2 + C_6) - C_3 C_6)\, G_{amp}) \\
d_1 =& - C_6\, ({C_3}^2 + 2\, C_6\, (C_4 + C_5 - C_5\, G_{amp}) + C_3\, (C_4 + C_5 + C_6 - C_5\, G_{amp} + C_6\, G_{amp})) - \\
& - C_2\, ({C_3}^2 + (C_1 + 4\, C_6)\, (C_4 + C_5 - C_5\, G_{amp}) + C_3\, (C_4 + C_5 - C_5\, G_{amp} + C_6\, (2 + G_{amp}))) - \\
& - C_1\, ({C_3}^2 + C_3\, (C_4 + C_5 - C_5\, G_{amp}) + C_6\, (C_4 + C_5 - C_5\, G_{amp})) - \\
& - {C_2}^2\, (C_3 + 2\, (C_4 + C_5 - C_5\, G_{amp})) \\
d_2 =& (C_1 + C_2 + C_6)\, (C_2 + C_3 + C_6)\, (C_3 + C_4 + C_5 - C_5\, G_{amp})
\end{aligned} \tag{5.11}$$

Using the ideal transfer function (5.10), the SC filter was designed to have a cutoff frequency f_c = 500 kHz for a clock frequency F_s = 50 MHz. The design used to achieve these specifications is shown in Table 5.1. The VCVS gain G_{amp} will be used to design the amplifier circuit in the next step.

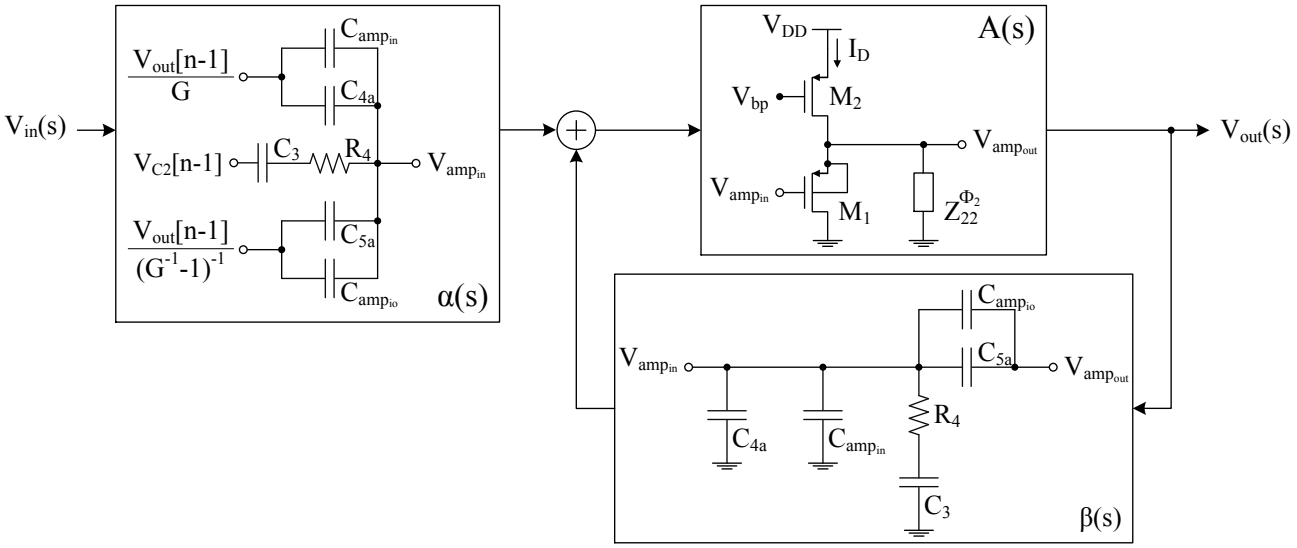
Table 5.1: Design used in the ideal lowpass SC filter.

C_1 (fF)	C_2 (fF)	C_3 (fF)	C_4 (fF)	C_5 (fF)	C_6 (fF)	G_{amp}	F_s (MHz)
360.0	1420.0	200.0	2130.0	274.5	6150.0	0.975	50.0

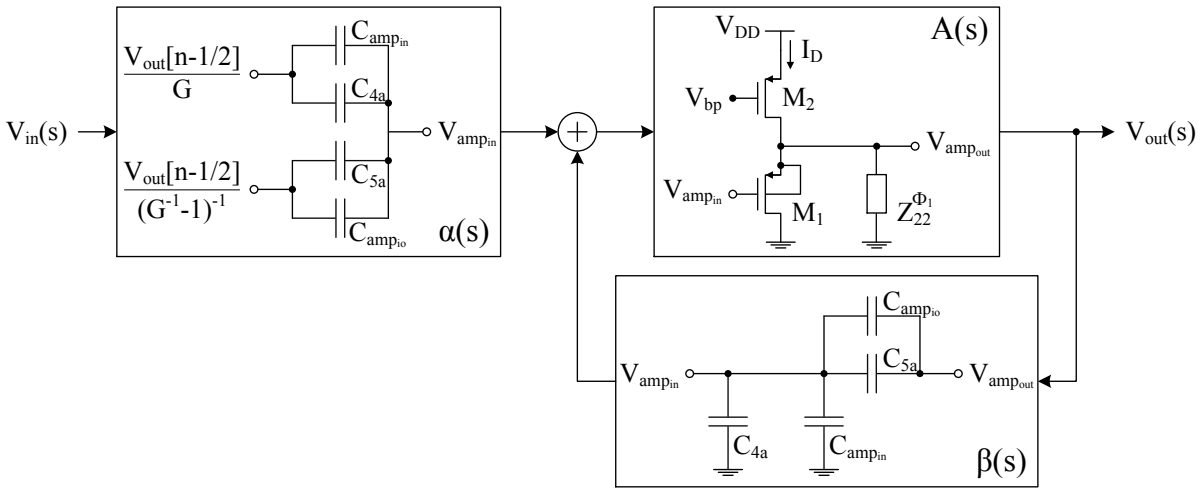
5.1.2.2 Design of the Amplifier

Fig. 5.4 shows the blocks used for the calculation of the closed-loop transfer function in each phase. For the sake of clarity, the filter is implemented with a simple common-drain amplifier, represented as block $A(s)$. The loads $Z_{22}^{\Phi_1}$ and $Z_{22}^{\Phi_2}$, which represent the filters' impedance seen by the output of the amplifier during phase Φ_1 and Φ_2, respectively, are given by (5.12), considering that the open switches have a resistance $R \approx \infty$.

$$\begin{aligned}
Z_{22}^{\Phi_1}\Big|_{R_2,\, R_4 \to \infty} &= \left(\left(\left(\frac{1}{s\, C_3} + R_3\right) \parallel \frac{1}{s\, C_2}\right) + \frac{1}{s\, C_6}\right) \parallel \frac{1}{s\, C_{5a}} \\
Z_{22}^{\Phi_2}\Big|_{R_1,\, R_3 \to \infty} &= \left(\left(\left(\frac{1}{s\, C_1} + R_2\right) \parallel \frac{1}{s\, C_2}\right) + \frac{1}{s\, C_6}\right) \parallel \frac{1}{s\, C_{5a}}
\end{aligned} \tag{5.12}$$



(a)



(b)

Figure 5.4: Closed-loop diagram during (a) phase Φ_2 and (b) phase Φ_1.

Capacitors $C_{amp_{in}}$ and $C_{amp_{io}}$ represent, respectively, the amplifier's input and input-to-output parasitic capacitances and are given by (5.13). These capacitances need to be considered in both feedback and input factors in order to obtain the correct signal amplitude. Capacitors C_{4a} and C_{5a} represent the compensated values of the filter capacitors C_4 and C_5, considering the effects of the amplifier's parasitic capacitances, in order to obtain the same frequency response as the one obtained from the ideal design.

$$\begin{aligned} C_{amp_{in}} &= C_{gd_1} & C_{amp_{io}} &= C_{gb_1} + C_{gs_1} \\ C_{4a} &= C_4 - C_{amp_{in}} & C_{5a} &= C_5 - C_{amp_{io}} \end{aligned} \tag{5.13}$$

The open-loop gain of the common-drain amplifier is given by (5.14).

$$A(s) = \frac{v_{amp_{out}}}{v_{amp_{in}}} = \frac{g_{m_1}\, Z_{22} + s\,(C_{bg_1} + C_{sg_1})\, Z_{22}}{1 + (g_{m_1} + g_{ds_1} + g_{ds_2})\, Z_{22} + s\,(C_{bd_1} + C_{bg_1} + C_{sd_1} + C_{sg_1} + C_{db_2} + C_{dg_2} + C_{ds_2})\, Z_{22}} \tag{5.14}$$

Since, in most cases, the capacitors in the filter have an initial charge different from zero (initial condition), this charge must also be treated as an input signal in the calculation of the input factors. These initial conditions can be calculated, from (5.9), for any given period, where $G_{amp} = A(0)$. Considering the circuits shown in Fig. 5.4, the input factors are given by

$$\begin{aligned}
\alpha^{\Phi_1}(s) &= \frac{1}{G_{amp}} - \frac{C_{5a} + C_{amp_{io}}}{C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}}}\, V_{out}[n - 1/2] \\
\alpha^{\Phi_2}(s) &= \frac{C_3}{C_3 + C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}} + s\,(C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}})\, R_4 C_3}\, V_{C2}[n] + \\
&+ \frac{\big(1 + R_4 C_3\, s\big)\big(C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}} - G_{amp}\,(C_{5a} + C_{amp_{io}})\big)}{G_{amp}\,\big(C_3 + C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}} + s\,(C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}})\, R_4 C_3\big)}\, V_{out}[n]
\end{aligned} \tag{5.15}$$

and the feedback factors are given by

$$\begin{aligned}
\beta^{\Phi_1}(s) &= \frac{C_{5a} + C_{amp_{io}}}{C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}}} \\
\beta^{\Phi_2}(s) &= \frac{(C_{5a} + C_{amp_{io}})(1 + R_4 C_3\, s)}{C_3 + C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}} + s\,(C_{4a} + C_{5a} + C_{amp_{in}} + C_{amp_{io}})\, R_4 C_3}
\end{aligned} \tag{5.16}$$

The closed-loop transfer function (A_{fb}) during phase Φ_1 and phase Φ_2, considering the effect of the amplifier, can be obtained using equation (5.2). Using the inverse Laplace transform, either directly or after decomposing A_{fb} into partial fractions, the time-domain behavior of the amplifier's output node can be estimated. To verify if the amplifier has the required gain value G_{amp}, obtained in the previous step, it is necessary to repeat the calculation of the closed-loop transfer function, using G_{amp} as the gain of the amplifier, considering that the closed switches have a resistance $R \approx 0$, and neglecting the influence of input and input-to-output parasitic capacitances, as shown by equation (5.17). If the amplifier has the desired gain, the voltage values of $v_{out}(t)$ and $v_{out_{ideal}}(t)$, at the end of each phase, will be identical.

$$A_{fb_{ideal}}(s) = \left.\frac{\alpha(s)\, G_{amp}}{1 - \beta(s)\, G_{amp}}\right|_{R,\, C_{amp_{in}},\, C_{amp_{io}} \to 0} \quad \Rightarrow \quad v_{out_{ideal}}(t) = \mathcal{L}^{-1}\left(\frac{1}{s}\, A_{fb_{ideal}}(s)\right) \tag{5.17}$$

Fig. 5.5 shows the time-domain behavior of the output signal, obtained from the closed-loop analysis method, using the ideal amplifier block ($v_{out_{ideal}}(t)$) and the common-drain amplifier ($v_{out}(t)$), and from a Spectre simulation, during the third sampling period. The comparison is made in this period to validate the initial conditions in the input factors, which are all non-zero after two periods.

These results show that the amplifier has the desired gain, since the voltage values at the end of each phase are identical, and that there is a reasonable degree of accuracy in the time-domain behavior of the signal, apart from the initial decrease in voltage. This effect is not predicted since the diagram shown in Fig. 5.4 does not account for the influence of every component in the circuit, i.e., both feedback and input factor do not account for the influence of the feedback path formed by capacitor C_6 since, due to the switches, it is never connected to the input of the amplifier, and this path is the one responsible for the initial decrease in voltage at the beginning of both phases.

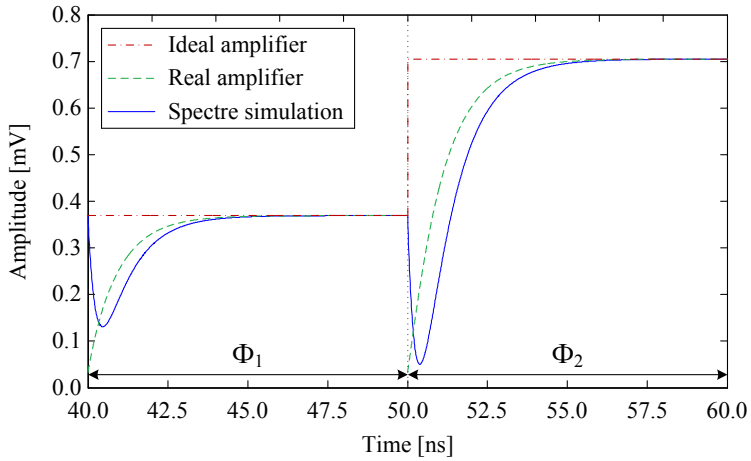


Figure 5.5: Time-domain comparison of the amplifier's output node between the closed-loop analysis method and the Spectre simulation of the filter during one sampling period.

The common-drain amplifier was implemented in a standard 130 nm CMOS technology. The design obtained for the amplifier, having the necessary gain G_{amp} and enough bandwidth, is shown in Table 5.2.

Table 5.2: Common-drain amplifier design and parameters.

W_1 (µm)	L_1 (µm)	W_2 (µm)	L_2 (µm)	I_D (µA)	C_{gb_1} (fF)	C_{gd_1} (fF)	C_{gs_1} (fF)
100.00	0.24	25.00	0.96	101.28	17.6	29.9	131.9

R_1 (kΩ)⋆	R_2 (kΩ)⋆	R_3 (kΩ)⋆	R_4 (kΩ)⋆	R_{open} (TΩ)⋆	V_{DD} (V)	V_{vcm} (V)	Power (µW)
1.25	1.00	1.50	1.75	1.00	1.20	0.45	255.07

⋆ In this step the switches are considered as ideal

5.1.2.3 Design of the Switches RC Time Constants

The switches RC time constants are calculated using first-order RC approximations with the circuits shown in Fig. 5.6, which are obtained by considering all switches in the active phase as short-circuits, except for the switch being analyzed, and all switches in the opposite phase as open-circuits. Note that, since the value of the switches parasitic capacitances change depending if the switch is ON or OFF, an approximate value is used in the equations, which is given by the average value between the ON and OFF values.

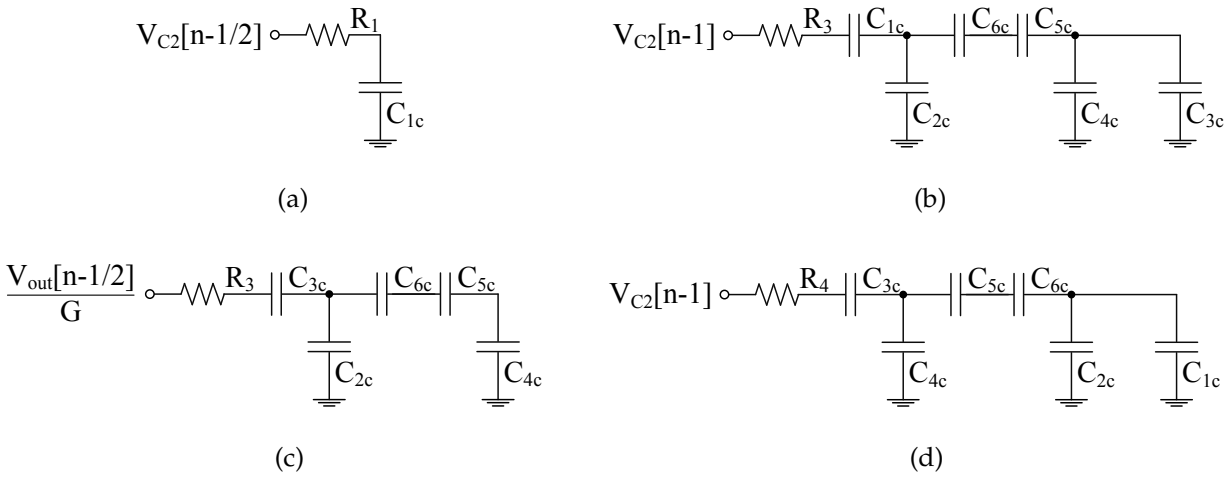


Figure 5.6: Circuit used to calculate the switched network time constant using first-order RC approximation for switch: (a) S_1 (Φ_1), (b) S_2 (Φ_2), (c) S_3 (Φ_1), and (d) S_4 (Φ_2).

where,

$$\begin{aligned}
C_{1c} &= C_1 - 0.5 \times (C_{ss_{on1}} + C_{ss_{off1}}) - 0.5 \times (C_{dd_{on2}} + C_{dd_{off2}}) \\
C_{2c} &= C_2 - 0.5 \times (C_{ss_{on2}} + C_{ss_{off2}}) - 0.5 \times (C_{dd_{on3}} + C_{dd_{off3}}) \\
C_{3c} &= C_3 - 0.5 \times (C_{ss_{on3}} + C_{ss_{off3}}) - 0.5 \times (C_{dd_{on4}} + C_{dd_{off4}}) \\
C_{4c} &= C_4 - C_{amp_{in}} - 0.5 \times (C_{ss_{on4}} + C_{ss_{off4}}) \\
C_{5c} &= C_5 - C_{amp_{io}} \\
C_{6c} &= C_6
\end{aligned} \tag{5.18}$$

The time constants in the switched networks (V_{C1c} and V_{C3c}) are obtained by multiplying the switches' resistance with the equivalent capacitance seen from the switch. Alternatively, the voltage in these nodes can be calculated by solving the system resulting from using Kirchhoff's current law (KCL) and calculating the poles of the transfer function to obtain the time constants, considering as input the charge stored in the switched network capacitor in the previous phase. Using either approach, the switches time constants are given by (5.19).

$$\begin{aligned}
\tau_{s1} &= R_1\, C_{1c} \\
\tau_{s2} &= R_2 \left[\left(\left((C_{3c} + C_{4c})^{-1} + C_{6c}^{-1} + C_{5c}^{-1} \right) \parallel C_{2c}^{-1} \right) + C_{1c}^{-1} \right]^{-1} \\
\tau_{s3} &= R_3 \left[\left(C_{4c}^{-1} + C_{5c}^{-1} + C_{6c}^{-1} \right) \parallel C_{2c}^{-1} \right) + C_{3c}^{-1} \right]^{-1} \\
\tau_{s4} &= R_4 \left[\left(\left((C_{1c} + C_{2c})^{-1} + C_{6c}^{-1} + C_{5c}^{-1} \right) \parallel C_{4c}^{-1} \right) + C_{3c}^{-1} \right]^{-1}
\end{aligned} \tag{5.19}$$

Table 5.3 shows the time it takes to reach a settling error below 1% in nodes V_{C1c} and V_{C3c}. The time-domain variation of the voltage in these nodes is shown in Fig. 5.7 to give a better perspective on the differences between the first-order approximation used and the Spectre simulation of the real filter.

Results show that the largest difference occurs in the first switched network (V_{C1c}) during phase Φ_2 and in the second switched network (V_{C3c}) during phase Φ_1, this is expected since during these phases there is a path to the output of the amplifier, through capacitor C_6, and the first-order approximation does not correctly model the effect of the amplifier's load effect.

Table 5.3: Comparison of the time taken to reach a settling error below 1 % between the first-order approximation used and the Spectre simulation.

Using equation (5.19)				Using Spectre simulation			
$t_{S1}^{1\%}$ (ns)	$t_{S2}^{1\%}$ (ns)	$t_{S3}^{1\%}$ (ns)	$t_{S4}^{1\%}$ (ns)	$t_{S1}^{1\%}$ (ns)	$t_{S2}^{1\%}$ (ns)	$t_{S3}^{1\%}$ (ns)	$t_{S4}^{1\%}$ (ns)
2.07	1.36	1.23	1.49	2.07	3.55	2.45	3.36

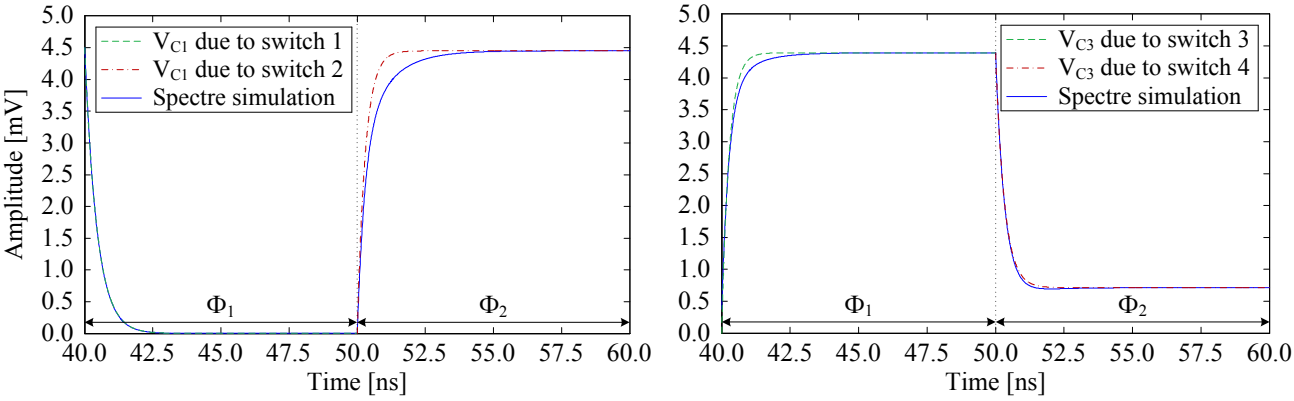


Figure 5.7: Time-domain comparison of the switches time constants between the first order RC approximations method and the Spectre simulation of the filter during one sampling period.

In Spectre, the frequency response is determined from the transient electrical simulation of the impulse response, obtained by applying a step with duration of one clock period ($1/F_s$) to the input node of the circuit. The output signal is then sampled at a rate of $1/F_s$, at the end of clock phase Φ_1. Finally, the frequency response is obtained by calculating the FFT of the sampled output signal. This procedure requires a long transient electrical simulation in order to obtain, at least, 10000 samples of the output voltage to calculate the FFT accurately.

The frequency response of the filter can also be determined using a faster periodic steady state (PSS)/periodic alternating current (PAC) analysis, this simulation requires adding an ideal sample-and-hold circuit at the output of the filter to obtain accurate results. Even in this case, it is less accurate than a transient simulation for frequencies close to, and higher than $F_s/2$, which might be critical for filters with bilinear type transfer functions or filters with a cutoff/center frequency close to $F_s/2$.

The frequency responses of the lowpass SC filter using ideal components and using real components are shown in Fig. 5.8. Results show that the amplifier was designed with the necessary gain value G_{amp} and has enough bandwidth for the design used in the filter, which is shown in Table 5.4. Since the filter is sensitive to the parasitic effects of the components, due to the architecture used and the absence of a virtual ground node, the parasitic capacitances were compensated into the design of the filter. The results also show that this compensation was correctly performed.

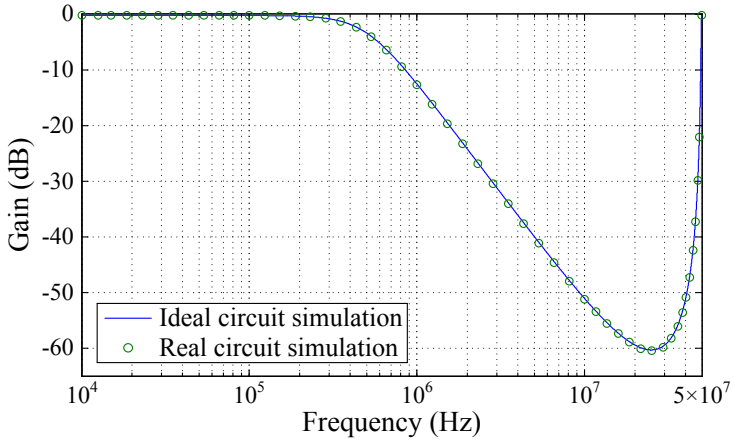


Figure 5.8: Frequency response comparison of the SC filter (ideal design Vs. real compensated design).

Table 5.4: Design used in the switches and compensated capacitor values.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)
358.0	1418.2	198.6	2129.3	125.0	6150.0

W_{s_1}/L_{s_1}	$R_{on_1}(k\Omega)$	W_{s_2}/L_{s_2}	$R_{on_2}(k\Omega)$	W_{s_3}/L_{s_3}	$R_{on_3}(k\Omega)$	W_{s_4}/L_{s_4}	$R_{on_4}(k\Omega)$
0.68/0.12	1.23	0.84/0.12	1.00	0.56/0.12	1.50	0.48/0.12	1.74

$C_{dd1_{on}}$ (fF)	$C_{ss1_{on}}$ (fF)	$C_{dd2_{on}}$ (fF)	$C_{ss2_{on}}$ (fF)	$C_{dd3_{on}}$ (fF)	$C_{ss3_{on}}$ (fF)	$C_{dd4_{on}}$ (fF)	$C_{ss4_{on}}$ (fF)
1.38	1.13	1.69	1.38	1.14	0.94	0.98	0.81

$C_{dd1_{off}}$ (fF)	$C_{ss1_{off}}$ (fF)	$C_{dd2_{off}}$ (fF)	$C_{ss2_{off}}$ (fF)	$C_{dd3_{off}}$ (fF)	$C_{ss3_{off}}$ (fF)	$C_{dd4_{off}}$ (fF)	$C_{ss4_{off}}$ (fF)
0.57	0.57	0.69	0.69	0.41	0.48	0.42	0.42

From this example, it is possible to conclude that this approach only gives approximate results. In the closed-loop transfer function, the influence of some components might not be included in the calculations, if there is no path from the component to the input of the amplifier, in a given phase, degrading the accuracy of the approximation. The switches time constants relies on a first-order approximations. This method might not give approximate results in nodes with a path to the amplifier's output, since its influence is modeled only by the input and input-to-output parasitic capacitances. In both methods, this means that it is necessary to consider the phases with a smaller length than the actual value, to ensure that complete settling is achieved in every node of the circuit, even with these unmodeled effects. This can lead to an oversizing in the amplifier's power and in the switches area.

A more accurate method, is to consider the equivalent circuits in each phase, at transistor level (using its medium frequency small-signal model), and calculate the time-domain voltages in the circuit. However, this relies on the calculation of the inverse Laplace transform of the frequency-domain voltages, which is not efficient due to the size of the expressions. In the next section, an accurate (numeric) method is presented, capable of calculating the time-domain voltages (and the discrete-time transfer function) without the need of the inverse Laplace transform, resulting in a more efficient and accurate method to be implemented in a computer.

5.2 Non-Hierarchical Numeric Methodology for the Analysis of SC Filters

5.2.1 Method Description

The non-hierarchical numeric methodology for the analysis of SC circuits is divided into three steps: the first step consists in the calculation of the time-domain voltage in each node of the circuit, for each clock phase, from the system of differential equations that describes the circuit's continuous-time behavior in any clock phase; the second step consists in converting the time-domain equations, obtained in the previous step, into discrete-time, by replacing the time variable with the duration of the respective clock phase and, combining them together into a single system of equations that completely describes the behavior of the circuit during one sampling period, based on the initial conditions at each node of the circuit; finally, the third step consists of obtaining the frequency response of the circuit from the discrete-time equations. Although the method is presented for two-phase circuits, it can be extended to circuits with more phases.

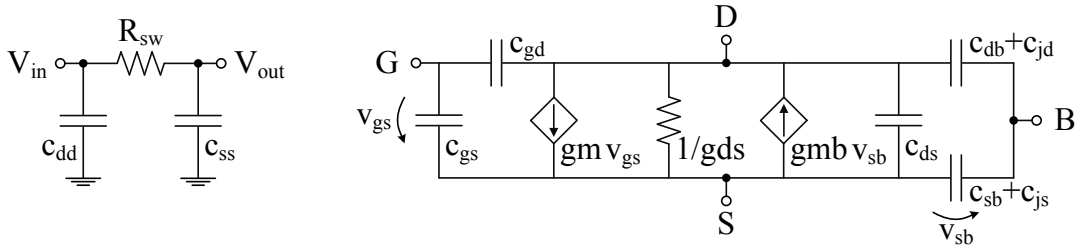
5.2.1.1 Obtaining the circuit's time-domain voltages

Using KCL, the time-domain behavior of a SC circuit can be described by a single $m \times m$ system of differential equations (5.20), where the transistors in the circuit are replaced by their equivalent linear model, which includes resistances, transconductances, and capacitances. This system is configured for any clock phase by setting the equivalent resistance values of the switches to either the ON or OFF values.

$$\begin{aligned}
G_{11}\, v_1 + C_{11}\, \frac{dv_1}{dt} + \cdots + G_{1m}\, v_m + C_{1m}\, \frac{dv_m}{dt} + G_1\, v_{in} + C_1\, \frac{dv_{in}}{dt} &= 0 \\
G_{21}\, v_1 + C_{21}\, \frac{dv_1}{dt} + \cdots + G_{2m}\, v_m + C_{2m}\, \frac{dv_m}{dt} + G_2\, v_{in} + C_2\, \frac{dv_{in}}{dt} &= 0 \\
&\vdots \\
G_{m1}\, v_1 + C_{m1}\, \frac{dv_1}{dt} + \cdots + G_{mm}\, v_m + C_{mm}\, \frac{dv_m}{dt} + G_m\, v_{in} + C_m\, \frac{dv_{in}}{dt} &= 0
\end{aligned} \tag{5.20}$$

where m represents the number of nodes in the circuit minus input and ground nodes, and where G_{xy} and C_{xy} represent, respectively, the total conductance and total capacitance between nodes x and y; G_x and C_x represent, respectively, the total conductance and total capacitance between node x and the circuit's input node.

The model considered for the switches, shown in Fig. 5.9(a), includes a resistor and two parasitic capacitances (the equivalent capacitance seen from the drain c_{dd} and the one seen from the source c_{ss}). The medium frequency model considered for the transistors is shown in Fig. 5.9(b). Note that, in some technologies, the influence of the junction capacitances (c_{jd} and c_{js}) may already be included in the capacitances connected to the substrate (c_{db}, c_{gb}, c_{sb}). The numerical values of gm, gds, gmb and the parasitic capacitances of the transistors are obtained using a DC operating point (OP) simulation of the circuit. The transistors operating as switches require two DC OP simulations to obtain the numerical values of the parameters when the switch is ON and when it is OFF.



(a) (b)

Figure 5.9: Transistor model: (a) linear region (switch), (b) saturation region (amplifier).

The resulting system of differential equations (5.20) includes all the effects of the transistors in the circuit, including those in the amplifier and in the switches.

The system (5.20) can be rewritten to the form shown in (5.21). Note that the capacitance and conductance matrices, and the input vectors can be directly obtained from inspection of the circuit's netlist.

$$-\underbrace{\begin{bmatrix} C_{11} & \dots & C_{1m} \\ \vdots & \ddots & \vdots \\ C_{m1} & \dots & C_{mm} \end{bmatrix}}_{\text{Capacitance Matrix}} \cdot \begin{bmatrix} \frac{dv_1}{dt} \\ \vdots \\ \frac{dv_m}{dt} \end{bmatrix} = \underbrace{\begin{bmatrix} G_{11} & \dots & G_{1m} \\ \vdots & \ddots & \vdots \\ G_{m1} & \dots & G_{mm} \end{bmatrix}}_{\text{Conductance Matrix}} \cdot \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix} + \begin{bmatrix} G_1 \\ \vdots \\ G_m \end{bmatrix} \cdot v_{in} + \begin{bmatrix} C_1 \\ \vdots \\ C_m \end{bmatrix} \cdot \frac{dv_{in}}{dt} \tag{5.21}$$

The system of differential equations can be simplified to the form shown in (5.22) by multiplying both sides of equation (5.21) by the inverse of the capacitance matrix.

$$\begin{aligned} \frac{dv_x}{dt} &= \sum_{i=1}^{m} a_{xi}\, v_i + b_x\, v_{in} + c_x\, \frac{dv_{in}}{dt} \qquad x = 1, 2, ..., m \\ \frac{dv}{dt} &= \boldsymbol{A}\, v + \boldsymbol{b}\, v_{in} + \boldsymbol{c}\, \frac{dv_{in}}{dt} \end{aligned} \tag{5.22}$$

where,

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{bmatrix} \qquad \begin{aligned} \boldsymbol{b} &= \begin{bmatrix} b_1 & b_2 & \dots & b_m \end{bmatrix}^T \\ \boldsymbol{c} &= \begin{bmatrix} c_1 & c_2 & \dots & c_m \end{bmatrix}^T \end{aligned} \tag{5.23}$$

Coefficients b_x and c_x in (5.22) are non-zero only if the input signal is connected through a resistor (switch) or through a capacitor to node x, respectively.

The time-domain voltage equations are obtained using the eigenvalues and eigenvectors of matrix $\boldsymbol{A}$, which are calculated from:

$$\text{Det}(\boldsymbol{A} - \lambda\, \boldsymbol{I}) = 0 \tag{5.24}$$

The eigenvalues λ are the roots of the polynomial obtained from (5.24), where $\boldsymbol{I}$ is the $m \times m$ identity matrix,

$$(\boldsymbol{A} - \lambda\, \boldsymbol{I})\, \boldsymbol{v} = 0 \tag{5.25}$$

The non-zero vector $\boldsymbol{v}$, for each eigenvalue, is an eigenvector of $\boldsymbol{A}$. If matrix $\boldsymbol{A}$ is non-defective, i.e, if the matrix has m linearly independent eigenvectors, each eigenvalue λ with associated eigenvector $\boldsymbol{v}$, in the form of $e^{\lambda t}\ \boldsymbol{v}$, will be a part of the solution of the system of differential equations (5.22). The eigenvectors are rearranged to form matrix $\boldsymbol{V_{ev}}$ where, in this case, each column represents an eigenvector of matrix $\boldsymbol{A}$.

$$\boldsymbol{V_{ev}}(t) = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1m} \\ v_{21} & v_{22} & \cdots & v_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & \cdots & v_{mm} \end{bmatrix} \tag{5.26}$$

If matrix $\boldsymbol{A}$ is defective[2] [78], the matrix will not have m linearly independent eigenvectors and will have repeated eigenvalues (λ_r). In this case, the solutions due to the repeated roots will be in the form:

$$\left(\sum_{i=1}^{y} \frac{t^{y-i}}{(y-i)!}\, \boldsymbol{v_i} \right) e^{\lambda_r t} \tag{5.27}$$

where y will change value for each repeated root, starting from one up to the multiplicity of the particular eigenvalue. Vector v_i represents the generalized eigenvectors of these roots, which are calculated sequentially using the linearly independent eigenvectors corresponding to the repeated roots.

$$\begin{gathered} (\mathbf{A} - \lambda_r\, \mathbf{I})\, v_1 = 0 \\ (\mathbf{A} - \lambda_r\, \mathbf{I})\, v_2 = v_1 \\ (\mathbf{A} - \lambda_r\, \mathbf{I})\, v_3 = v_2 \\ \vdots \\ (\mathbf{A} - \lambda_r\, \mathbf{I})\, v_y = v_{y-1} \end{gathered} \tag{5.28}$$

The columns of matrix $\boldsymbol{V_{ev}}$ (5.26), corresponding to the eigenvectors of the repeated roots, will hold the vectors obtained from calculating the sum in (5.27).

The exponential of the eigenvalues, which are present in the solution regardless of whether matrix $\boldsymbol{A}$ is defective or not, are arranged to form matrix $\boldsymbol{V_\lambda}$.

$$\boldsymbol{V_\lambda}(t) = \begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_m t} \end{bmatrix} \tag{5.29}$$

[2] Since a small perturbation in matrix $\boldsymbol{A}$ is sufficient to split the repeated eigenvalues, a small variation can be added to at least one of the elements of the circuit, avoiding the need to calculate generalized eigenvectors and the increased complexity of matrix $\boldsymbol{V_{ev}}$ due to the repeated eigenvalues λ_r (5.27).

To obtain the forced response, i.e., the influence of vectors $\boldsymbol{b}$ and $\boldsymbol{c}$ on each node of the circuit, it is necessary to calculate vector $\boldsymbol{v_p}$ (5.30), which represents the particular solutions of the system. Since the goal is to calculate the voltages in the circuit when the switches change state (difference equations), this corresponds to considering the input signal $v_{in}(t)$ as a Heaviside step function.

$$\begin{gathered}\frac{dv}{dt} = 0 \;\Leftrightarrow\; \boldsymbol{A}\, v + \boldsymbol{b}\, v_{in} + \boldsymbol{c}\, \tfrac{dv_{in}}{dt} = 0 \;\Rightarrow\; \boldsymbol{v_p} = -\boldsymbol{A}^{-1}.\, \boldsymbol{b} \\ \boldsymbol{v_p} = \begin{bmatrix} \rho_1 & \rho_2 & \dots & \rho_m \end{bmatrix}^T\end{gathered} \tag{5.30}$$

The matrix of constants $\boldsymbol{V_{k_{nodes}}}$ (5.31), which results from the homogeneous response, and the vector of constants $\boldsymbol{v_{k_{input}}}$ (5.32), which results from combining the forced and homogenous responses, are obtained using matrix $\boldsymbol{V_{ev}}$ and vectors $\boldsymbol{c}$ and $\boldsymbol{v_p}$ (V_{ic_x} is the initial condition at node x, where x represents the number of the node, according to which the system of equations in (5.20) was built).

$$\boldsymbol{V_{k_{nodes}}} = \boldsymbol{V_{ev}}(0)^{-1} \,.\, \underbrace{\begin{bmatrix} V_{ic_1} & 0 & \cdots & 0 \\ 0 & V_{ic_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & V_{ic_m} \end{bmatrix}}_{\text{initial conditions at each node}} \tag{5.31}$$

$$\boldsymbol{v_{k_{input}}} = \boldsymbol{V_{ev}(0)^{-1}} \,.\, (\boldsymbol{c} - \boldsymbol{v_p})\, \boldsymbol{V_{in}} \tag{5.32}$$

The time-domain voltages for every node in the circuit can be calculated from (5.26), (5.29), (5.30), (5.31), and (5.32) using:

$$\begin{aligned}\boldsymbol{V}(t) &= \boldsymbol{V_{ev}}(t) \,.\, \boldsymbol{V_\lambda}(t) \,.\, \boldsymbol{V_{k_{nodes}}} + \boldsymbol{V_{ev}}(t) \,.\, \boldsymbol{V_\lambda}(t) \,.\, \boldsymbol{V_{k_{input}}} + \boldsymbol{v_p} \,.\, \boldsymbol{V_{in}} \\ &= \boldsymbol{C} \,.\, \begin{bmatrix} V_{ic_1} & 0 & \cdots & 0 \\ 0 & V_{ic_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & V_{ic_m} \end{bmatrix} + \boldsymbol{d} \,.\, \boldsymbol{V_{in}}\end{aligned} \tag{5.33}$$

where matrix $\boldsymbol{C}$ and vector $\boldsymbol{d}$ are obtained from simplifying (5.33). Note that when vector $\boldsymbol{c}$ in (5.22) is non-zero, which occurs when a capacitor is connected directly to the input signal, it is necessary to consider the effect of the derivative of the input signal (considered as a step input) both at the beginning and at the end of the clock phases. This corresponds to adding two Dirac impulses (one positive and one negative), since the derivative of a step is a Dirac impulse.

Equation (5.33) can also be obtained using the matrix exponential method (which is mathematically equivalent to the method previously described), where $\boldsymbol{V}(t)$ is given by:

$$\boldsymbol{V}(t) = \underbrace{e^{\boldsymbol{A}\,t}\,.\begin{bmatrix} V_{ic_1} & 0 & \cdots & 0 \\ 0 & V_{ic_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & V_{ic_m} \end{bmatrix}}_{\text{homogeneous solution}} + \underbrace{\left(\int_0^t e^{\boldsymbol{A}\,(t-\tau)}.\,\boldsymbol{b}\,d\tau + e^{\boldsymbol{A}\,t}.\,\boldsymbol{c}\right)\boldsymbol{V_{in}}}_{\text{non-homogeneous solution}} \tag{5.34}$$

where,

$$\int_0^t e^{\boldsymbol{A}\,(t-\tau)}.\,\boldsymbol{b}\,d\tau = \left(e^{\boldsymbol{A}\,t} - \boldsymbol{I}\right).\,\boldsymbol{A}^{-1}.\,\boldsymbol{b} \tag{5.35}$$

5.2.1.2 Obtaining the circuit's difference equations

The voltage at the end of each clock phase is obtained by replacing t in (5.33) by the duration of the respective clock phase. In the case of a switched circuit with two phases, t is approximately half the sampling period ($T_s/2$).

$$V_x\left(\frac{T_s}{2}\right) = V_x[n-1/2] = \sum_{i=1}^{m} \boldsymbol{C}_{xi}\,V_i[n-1] + \boldsymbol{d}_x\,V_{in}[n-1] \tag{5.36}$$

Each clock phase defines a set of difference equations in which the initial conditions are the end voltage values obtained in the previous phase. The timing notation, shown in Fig. 5.10, is set considering that the signal is sampled at the end of clock phase Φ_1.

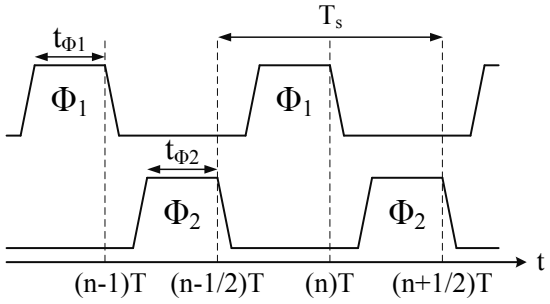


Figure 5.10: Non-overlapping clock phases.

$$\begin{aligned} \boldsymbol{V}[n-1/2] &= \boldsymbol{C}^{\boldsymbol{\Phi_2}}\,.\,\boldsymbol{V}[n-1] + \boldsymbol{d}^{\boldsymbol{\Phi_2}}\,V_{in}[n-1], \quad \text{phase 2} \\ \boldsymbol{V}[n] &= \boldsymbol{C}^{\boldsymbol{\Phi_1}}\,.\,\boldsymbol{V}[n-1/2] + \boldsymbol{d}^{\boldsymbol{\Phi_1}}\,V_{in}[n-1/2], \quad \text{phase 1} \end{aligned} \tag{5.37}$$

In order to describe the behavior of the circuit during a complete clock cycle the two previous sets of equations must be combined into a single system of equations. Equation (5.38) is the result of combining both clock phases into a single system of equations.

$$\boldsymbol{V}[n] = \boldsymbol{E}\,.\,\boldsymbol{V}[n-1] + \boldsymbol{h}\,V_{in}[n-1] + \boldsymbol{d}^{\boldsymbol{\Phi_1}}\,V_{in}[n-1/2] \tag{5.38}$$

where,

$$\boldsymbol{E} = \boldsymbol{C}^{\boldsymbol{\Phi_1}}.\,\boldsymbol{C}^{\boldsymbol{\Phi_2}} \quad \text{and} \quad \boldsymbol{h} = \boldsymbol{C}^{\boldsymbol{\Phi_1}}.\,\boldsymbol{d}^{\boldsymbol{\Phi_2}} \tag{5.39}$$

5.2.1.3 Obtaining the circuit's discrete-time transfer function

From the set of difference equations (5.38), it is possible to calculate the discrete-time transfer function from the input to any node of the circuit, $H(z) = num(z)/den(z)$. The denominator of the discrete-time transfer function, which is independent of the output node, can be calculated from:

$$den(z) = \mathrm{Det}(\boldsymbol{E} - z\,\boldsymbol{I}) \tag{5.40}$$

The numerator of the discrete-time transfer function depends on the desired output node j and can be calculated from (5.41), where $\mathrm{Det}(\boldsymbol{E} - z\,\boldsymbol{I})_{ij}$ is the determinant of the matrix obtained by removing line i and column j.

$$num_{V_j}(z) = \sum_{i=1}^{m} (-1)^{j+(i-1)} (\boldsymbol{h}_i + \boldsymbol{d}_i^{\boldsymbol{\Phi_1}}\, z^{1/2})\, \mathrm{Det}(\boldsymbol{E} - z\,\boldsymbol{I})_{ij} \tag{5.41}$$

Although the previously described procedure requires only numerical operations to obtain $H(z)$ for a specific frequency $\left(z = e^{j\,2\pi\,f_i\,T_s}\right)$, it requires repeating the calculation of equations (5.40) and (5.41) (which include computing matrix determinants) for each frequency value. An alternative method, which only requires making matrix computations once, for all the required frequency values, is to calculate the numerical values of the poles and zeros of $H(z)$. The poles are obtained by calculating the eigenvalues of matrix $\boldsymbol{E}$ and the zeros are obtained by calculating the generalized eigenvalues [79] (5.42). Note that depending on the line and column removed, the coefficients of z may no longer be in the diagonal of the matrix and the gain factor to obtain the numerator's characteristic polynomial is no longer 1.

$$\mathrm{Det}(\boldsymbol{E} - z\,\boldsymbol{I})_{ij} = \mathrm{Det}(\boldsymbol{A}^{\dagger} - z\,\boldsymbol{B}^{\dagger}) \tag{5.42}$$

The gain factor is obtained from analyzing where the non-zero coefficients (1's) are located in matrix $B^{\dagger}$. When $i = j$ the 1's are in the diagonal of $\boldsymbol{B}^{\dagger}$, i.e., $\boldsymbol{B}^{\dagger} = \boldsymbol{I}$ (typical eigenvalue case), and the gain factor is 1. When $i \neq j$ the gain factor is a coefficient of matrix $\boldsymbol{A}^{\dagger}$ in the same position of the coefficient of matrix $\boldsymbol{B}^{\dagger}$ that does not have any 1's in its line and column.

It is important to note that, although the proposed non-hierarchical method is very accurate, independently of the circuit's settling condition, if charge equilibrium is not achieved at the end of either phase and, if the phase's non-overlapping time is not much smaller than the circuit's time constants, it is necessary to consider an additional phase, when both switches are OFF, in order to account for the charge variation in this state.

5.2.2 Example I: First-Order Passive SC Filter

In this section, a simple first-order passive SC filter, shown in Fig. 5.11, is used to demonstrate the analysis method presented in Section 5.2.1, considering the resistive effect of the switches.

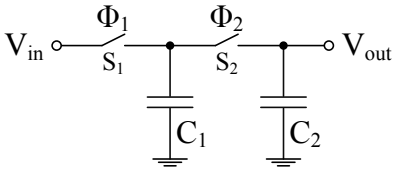


Figure 5.11: First-order passive SC filter.

5.2.2.1 Obtaining the circuit's discrete-time transfer function

Considering the equivalent circuit, shown in Fig. 5.12, which for the sake of clarity neglects the switches parasitic capacitances, the system of differential equations obtained by inspection is given by (5.43), where $G_i = 1/R_i$.

$$-\begin{bmatrix} C_1 & 0 \\ 0 & C_2 \end{bmatrix} \cdot \begin{bmatrix} \frac{dv_1}{dt} \\ \frac{dv_2}{dt} \end{bmatrix} = \begin{bmatrix} G_1+G_2 & -G_2 \\ -G_2 & G_2 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} -G_1 \\ 0 \end{bmatrix} v_{in} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \frac{dv_{in}}{dt} \tag{5.43}$$

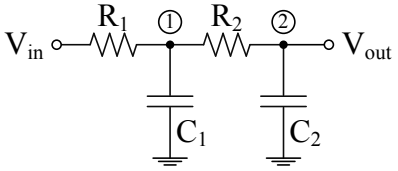


Figure 5.12: First-order passive SC filter considering the switches as resistors.

From (5.43) it is possible to obtain matrix $\boldsymbol{A}$ and vectors $\boldsymbol{b}$ and $\boldsymbol{c}$.

$$\boldsymbol{A} = \begin{bmatrix} -\dfrac{G_1+G_2}{C_1} & \dfrac{G_2}{C_1} \\ \dfrac{G_2}{C_2} & -\dfrac{G_2}{C_2} \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} \dfrac{G_1}{C_1} \\ 0 \end{bmatrix}, \qquad \boldsymbol{c} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{5.44}$$

Using equations (5.26), (5.29), (5.30), (5.31), and (5.32) with the matrices shown in (5.44), the time-domain voltages can be calculated, resulting in (5.45) for phase Φ_2 and in (5.46) for phase Φ_1. These equations are simplified considering that in a practical SC circuit, when the switch is open, $G_i \approx 0$.

$$\begin{aligned} \lim_{G_1 \to 0} V_1[n-1/2] &= \frac{C_1}{C_1+C_2} V_1[n-1] + \frac{C_2}{C_1+C_2} V_2[n-1] + \frac{C_2\, e^{-\alpha\, t^{\Phi_2}}}{C_1+C_2} \big(V_1[n-1] - V_2[n-1]\big) \\ \lim_{G_1 \to 0} V_2[n-1/2] &= \frac{C_1}{C_1+C_2} V_1[n-1] + \frac{C_2}{C_1+C_2} V_2[n-1] + \frac{C_1\, e^{-\alpha\, t^{\Phi_2}}}{C_1+C_2} \big(V_2[n-1] - V_1[n-1]\big) \end{aligned} \tag{5.45}$$

$$\begin{aligned}\lim_{G_2\to 0} V_1[n] &= e^{-\beta\, t^{\Phi_1}}(V_1[n-1/2]-V_{in}[n-1/2])+V_{in}[n-1/2]\\ \lim_{G_2\to 0} V_2[n] &= V_2[n-1/2]\end{aligned} \tag{5.46}$$

where,

$$\alpha=\frac{C_1+C_2}{C_1C_2}\,G_2,\qquad \beta=\frac{G_1}{C_1},\qquad t^{\Phi_1}=t^{\Phi_2}\approx T/2 \tag{5.47}$$

Using equations (5.45) and (5.46) to calculate the nodal voltages at the end of each clock phase and using the procedure described by (5.37), (5.38), and (5.39), the system of difference equations which completely describes the discrete-time behavior of the circuit can be obtained, where $\boldsymbol{C^{\Phi_1}}$ and $\boldsymbol{d^{\Phi_1}}$ are given by (5.48), $\boldsymbol{C^{\Phi_2}}$ and $\boldsymbol{d^{\Phi_2}}$ are given by (5.49), and matrix $\boldsymbol{E}$ is given by (5.50).

$$\boldsymbol{C^{\Phi_1}}=\begin{bmatrix} e^{-\beta\frac{T}{2}} & 0\\ 0 & 1\end{bmatrix},\qquad \boldsymbol{d^{\Phi_1}}=\begin{bmatrix}1-e^{-\beta\frac{T}{2}}\\ 0\end{bmatrix} \tag{5.48}$$

$$\boldsymbol{C^{\Phi_2}}=\begin{bmatrix}\dfrac{C_1+C_2\,e^{-\alpha\frac{T}{2}}}{C_1+C_2} & \dfrac{C_2-C_2\,e^{-\alpha\frac{T}{2}}}{C_1+C_2}\\ \dfrac{C_1-C_1\,e^{-\alpha\frac{T}{2}}}{C_1+C_2} & \dfrac{C_2+C_1\,e^{-\alpha\frac{T}{2}}}{C_1+C_2}\end{bmatrix},\qquad \boldsymbol{d^{\Phi_2}}=\begin{bmatrix}0\\0\end{bmatrix} \tag{5.49}$$

$$\boldsymbol{E}=\begin{bmatrix}\dfrac{e^{-\beta\frac{T}{2}}\left(C_1+C_2\,e^{-\alpha\frac{T}{2}}\right)}{C_1+C_2} & \dfrac{e^{-\beta\frac{T}{2}}\left(C_2-C_2\,e^{-\alpha\frac{T}{2}}\right)}{C_1+C_2}\\ \dfrac{C_1-C_1\,e^{-\alpha\frac{T}{2}}}{C_1+C_2} & \dfrac{C_2+C_1\,e^{-\alpha\frac{T}{2}}}{C_1+C_2}\end{bmatrix},\qquad \boldsymbol{h}=\begin{bmatrix}0\\0\end{bmatrix} \tag{5.50}$$

Using equations (5.40) and (5.41), the denominator (5.51) and numerator (5.52) of the discrete-time transfer function from the input to node V_2 can be obtained. The denominator of the transfer function $H_2(z)$ is obtained by calculating the determinant in (5.51) (or the poles of $H_2(z)$ by calculating the eigenvalues of matrix $\boldsymbol{E}$). The numerator of $H_2(z)$ is obtained by calculating the submatrices determinants in (5.52) (or the zeros of $H_2(z)$ by calculating the generalized eigenvalues of the submatrices).

$$Den(z)=\begin{vmatrix}\dfrac{e^{-\beta\frac{T}{2}}\left(C_1+C_2\,e^{-\alpha\frac{T}{2}}\right)}{C_1+C_2}-z & \dfrac{e^{-\beta\frac{T}{2}}\left(C_2-C_2\,e^{-\alpha\frac{T}{2}}\right)}{C_1+C_2}\\ \dfrac{C_1-C_1\,e^{-\alpha\frac{T}{2}}}{C_1+C_2} & \dfrac{C_2+C_1\,e^{-\alpha\frac{T}{2}}}{C_1+C_2}-z\end{vmatrix} \tag{5.51}$$

$$Num_{V_2}(z)=(-1)^2\,\gamma_1\,|\boldsymbol{E}-z\,\boldsymbol{I}|_{12}+(-1)^3\,\gamma_2\,|\boldsymbol{E}-z\,\boldsymbol{I}|_{22} \tag{5.52}$$

where,

$$\begin{aligned}\gamma_1 &= \boldsymbol{h}_1+\boldsymbol{d}_1^{\boldsymbol{\Phi_1}}\,z^{1/2}=\left(1-e^{-\beta\frac{T}{2}}\right)z^{1/2}\\ \gamma_2 &= \boldsymbol{h}_2+\boldsymbol{d}_2^{\boldsymbol{\Phi_1}}\,z^{1/2}=0\end{aligned} \tag{5.53}$$

By dividing the numerator (5.52) and the denominator (5.51), the discrete-time transfer function is given by (5.54).

$$H_2(z) = \frac{C_1\left(-1+e^{\alpha\frac{T}{2}}\right)\left(-1+e^{\beta\frac{T}{2}}\right)z^{1/2}}{C_1\left(-1+e^{\alpha\frac{T}{2}}\,z\right)\left(-1+e^{\beta\frac{T}{2}}\,z\right)+C_2(z-1)\left(-1+e^{(\alpha+\beta)\frac{T}{2}}\,z\right)} \tag{5.54}$$

Note that, although the equations in this section are presented symbolically, all the calculations can be performed numerically.

5.2.2.2 Validation test: Discrete-time transfer function Vs. Circuit simulation in Spectre

To verify the accuracy of the discrete-time transfer function (5.54), the first-order passive SC filter from Fig. 5.11 was simulated in Spectre with C_1 = 129.7 fF, C_2 = 2 pF, and clock frequency F_s = 100 MHz.

Fig. 5.13 presents the comparison between the frequency response of the circuit obtained using the proposed non-hierarchical method and using a Spectre electrical simulation for different switch resistance values (R_1 and R_2), showing that the numerical method accurately predicts the frequency response, even in the case of incomplete settling in the SC circuit. This allows the designer to accurately and efficiently determine when the size of the switches in the circuit become small enough to affect the desired frequency response of the SC filter due to incomplete settling in the circuit.

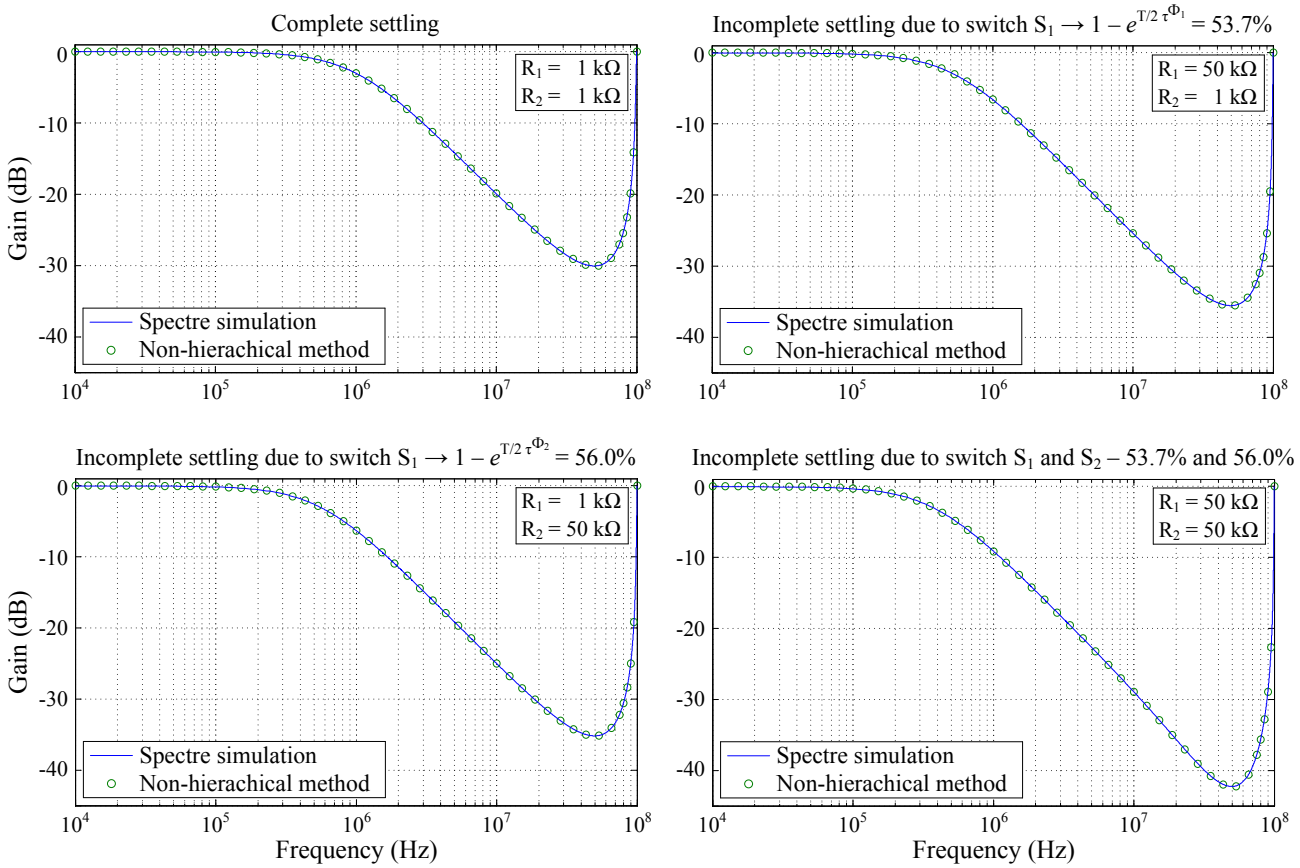


Figure 5.13: Frequency response comparison between the Spectre simulation and the proposed non-hierarchical analysis methodology for different values of switch resistance in the circuit of Fig. 5.11.

5.2.3 Example II: Second-Order Bandpass SC Filter

A second-order bandpass SC filter, shown in Fig. 5.14, is used to demonstrate the analysis method presented in Section 5.2.1, considering the non-ideal behavior of the switches and of the amplifier circuit. The filter architecture is based on the continuous-time bandpass Sallen-Key topology [35], replacing the resistors with parallel SC branches [80]. Capacitors $C_{p1,p2,p4,p5,p7}$ represent the equivalent parasitic capacitances from the switches (transistors) and the main capacitors in each node of the circuit. This architecture is capable of implementing bandpass filtering functions using low gain amplifiers with gain close to, but higher, than unity. For simplicity purposes the circuit is shown in single-ended configuration. In differential configuration capacitors $C_{1,2,4,5,7}$ are implemented with two capacitors in anti-parallel.

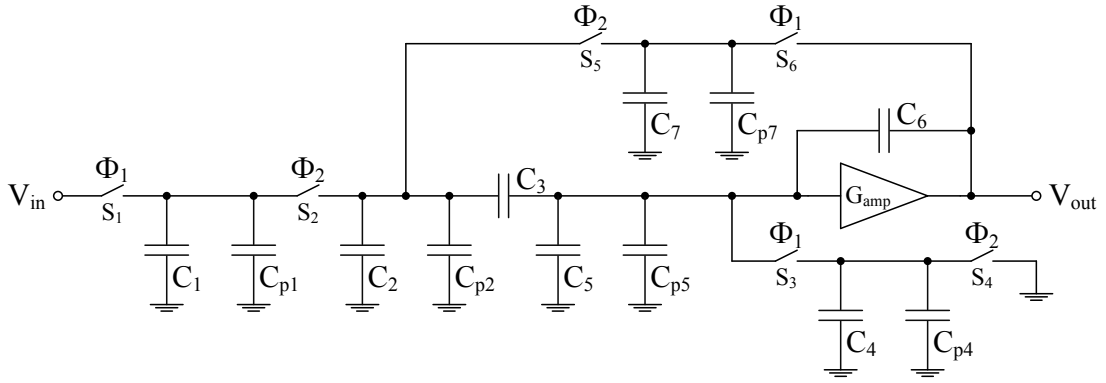


Figure 5.14: Second-order bandpass SC filter.

The amplifier circuit used in the bandpass SC filter is shown in Fig. 5.15. Transistors M_1 form a differential pair with source degeneration (transistors M_5), which are used to increase the amplifier's linearity in the output voltage range. Transistor M_2 is connected in common-drain configuration, setting the small signal impedance of the output node to $1/gm_2$ and forming a low gain voltage combiner together with transistor M_1. By applying the inverted input signal to the gate of M_2, this structure is capable of achieving more gain without sacrificing the GBW or increasing the size of the bias current of transistor M_1. Transistors M_6 and M_7 form a source-follower circuit that is used as a DC level shifter, allowing the input and output common-mode voltage to have the same value.

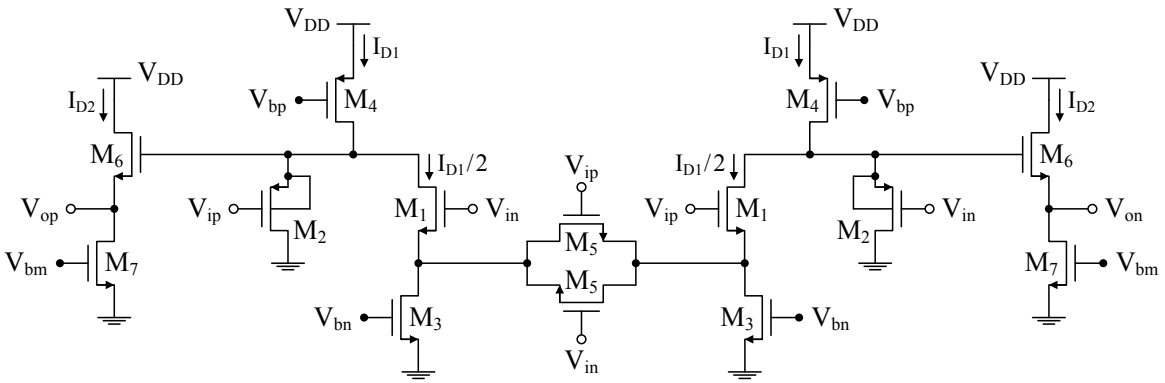


Figure 5.15: Differential voltage-combiner amplifier with source degeneration with a DC level shifter.

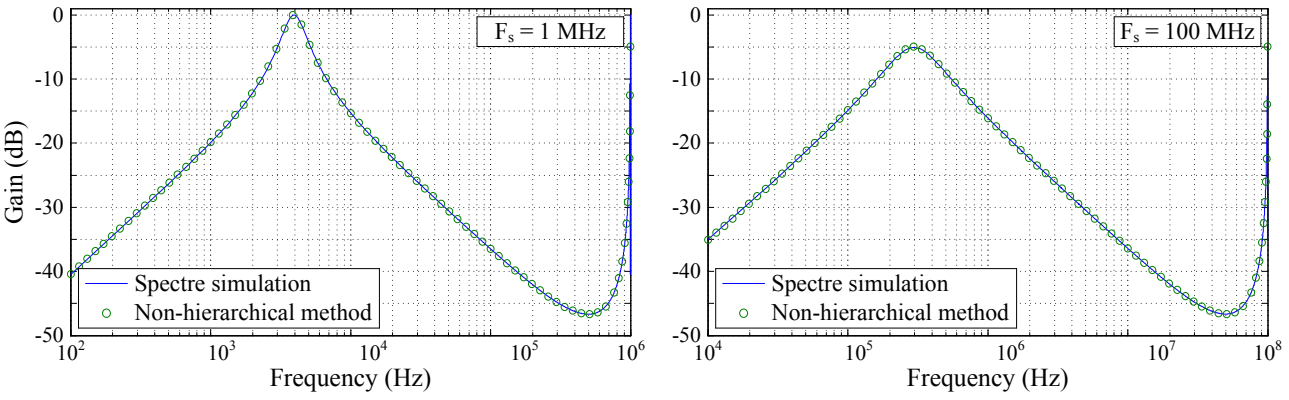
The second-order bandpass SC filter of Fig. 5.14 was implemented in a standard 130 nm CMOS technology. The filter was designed, using the same approach as in [80], to have a central frequency f_c = 4 kHz and a quality factor Q_p = 8/3 for a clock frequency F_s = 1 MHz, which resulted in the capacitor values shown in Table 5.5 and an amplifier gain G_{amp} = 1.42. The amplifier was designed to have the necessary gain G_{amp} and enough bandwidth to charge the capacitors in each clock phase. The design parameters of the bandpass SC filter are shown in Table 5.5. The switches used in the filter were implemented using NMOS transistors that are driven by regular clock phases ($V_{\Phi_1} = V_{\Phi_2} = V_{DD}$). The switches dimensions were chosen to ensure that every node in the circuit has complete settling for a clock frequency F_s = 1 MHz. Note that, the value used for the switches' parasitic capacitances is also obtained by calculating the average value between the ON and OFF values.

Table 5.5: Design used in the bandpass SC filter

C_1 (fF)	C_2 (fF)	C_3 (fF)	C_4 (fF)	C_5 (fF)	C_6 (fF)	C_7 (fF)
24.6	3062.8	8500.0	24.6	149.6	150.0	174.6
W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	W_6 (µm)	W_7 (µm)
8.00	185.00	44.00	54.00	46.00	101.00	99.00
L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	L_6 (µm)	L_7 (µm)
0.48	0.84	1.56	0.24	1.56	0.36	1.80

$W_{S1...S7}$ (µm)	$L_{S1...S7}$ (µm)	I_{D1} (µA)	I_{D2} (µA)	V_{DD} (mV)	V_{cm} (mV)	F_s (MHz)
1.00	0.12	88.0	58.0	900.0	450.0	1 - 100

The filter's frequency response was calculated and simulated for two different clock frequencies F_s. Fig. 5.16(a) shows the comparison between the filter's frequency response obtained using a Spectre simulation and the non-hierarchical analysis methodology for a clock frequency of 1 MHz. In this case, the time constants of each node are much lower than the duration of the clock phases, i.e., there is complete settling in the circuit. Fig. 5.16(b) shows the same comparison, but for a clock frequency of 100 MHz. In this case there is incomplete settling, since some of the time constants are higher than the duration of the clock phases. In both cases, results show that the proposed analysis methodology provides results that are in good agreement (below 0.1 dB difference for the range of frequencies observed) with the Spectre transient simulation.



(a) (b)

Figure 5.16: Frequency response comparison between the Spectre simulation and the proposed non-hierarchical analysis methodology of the bandpass filter with (a) complete and (b) incomplete settling.

For comparison purposes, the Spectre transient simulation (10000 samples) of the bandpass SC filter, with conservative accuracy, takes approximately 260 seconds using a single core of the Intel Xeon X5570 processor (2.93 GHz) in a shared, i.e., multi-user server; a PSS/PAC simulation takes 100 seconds for the same number of points; while obtaining the frequency response in the same machine using the presented method, including the two DC simulations, takes less than 200 milliseconds.

5.2.4 Example III: Second-Order Lowpass SC Filter

In this section, the second-order lowpass SC filter used as an example in Section 5.1.2, reproduced in Fig. 5.17 for convenience, is used to compare the accuracy of both methods presented in this chapter.

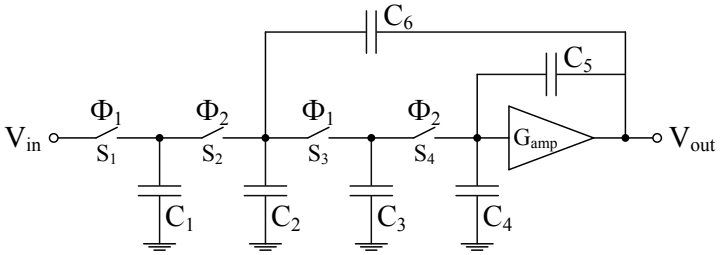


Figure 5.17: Second-order lowpass SC filter.

Considering the equivalent circuit, shown in Fig. 5.18, the system of differential equations used to calculate the time-domain voltages for any node in the circuit is given by (5.55).

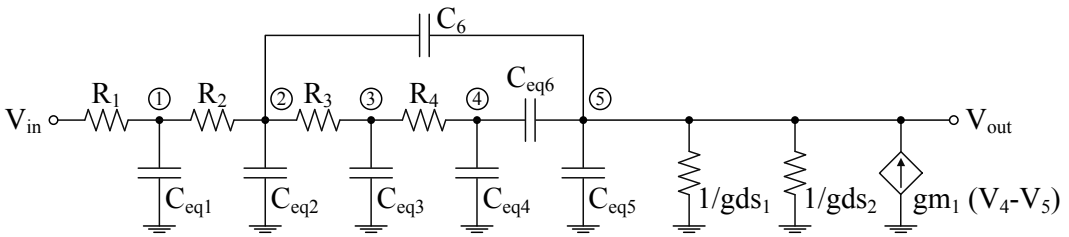


Figure 5.18: Second-order lowpass SC filter considering the switches as resistors and the amplifier as its equivalent medium frequency small signal model.

$$
\begin{aligned}
-\begin{bmatrix} C_{eq1} & 0 & 0 & 0 & 0 \\ 0 & C_{eq2}+C_6 & 0 & 0 & -C_6 \\ 0 & 0 & C_{eq3} & 0 & 0 \\ 0 & 0 & 0 & C_{eq4}+C_{eq6} & -C_{eq6} \\ 0 & -C_6 & 0 & -C_{eq6} & C_{eq5}+C_{eq6}+C_6 \end{bmatrix} \cdot \begin{bmatrix} \frac{dv_1}{dt} \\ \frac{dv_2}{dt} \\ \frac{dv_3}{dt} \\ \frac{dv_4}{dt} \\ \frac{dv_5}{dt} \end{bmatrix} = \\
= \begin{bmatrix} G_1+G_2 & -G_2 & 0 & 0 & 0 \\ -G_2 & G_2+G_3 & -G_3 & 0 & 0 \\ 0 & -G_3 & G_3+G_4 & -G_4 & 0 \\ 0 & 0 & -G_4 & G_4 & 0 \\ 0 & 0 & 0 & -gm_1 & gm_1+gds_1+gds_2 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} + \begin{bmatrix} -G_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} v_{in}
\end{aligned}
\tag{5.55}
$$

where,

$$C_{eq1} = C_1 + C_{p1} \qquad C_{eq2} = C_2 + C_{p2} \qquad C_{eq3} = C_3 + C_{p3} \qquad C_{eq4} = C_4 + C_{gd1} + C_{p4}$$
$$C_{eq5} = C_5 + C_{gb1} + C_{gs1} \qquad C_{eq6} = C_{bd1} + C_{sd1} + C_{jd1} + C_{db2} + C_{dg2} + C_{ds2} + C_{jd2} \tag{5.56}$$

and, $G_i = 1/R_i$ and capacitors $C_{p1,p2,p3,p4}$ represent the equivalent parasitic capacitances from the switches (transistors) and the main capacitors in each node of the circuit. Following the steps described in Section 5.2.1.1, the time-domain voltages in every node of the circuit can be calculated.

5.2.4.1 Accuracy test: Hierarchical Vs. Non-hierarchical methodology

Fig. 5.19 shows the time-domain behavior of nodes V_1 (V_{C1}), V_3 (V_{C3}), and V_5 (V_{out}), during the third sampling period, using the two analysis methodologies presented in this chapter. Results show that the proposed non-hierarchical numeric methodology delivers very accurate results when compared with the Spectre simulation, since the system of differential equations used includes all the effects of the transistors in the circuit, using only approximated values for the switches' parasitic capacitances since their value change depending if the switch is ON or OFF. The hierarchical symbolic methodology delivers less accurate results since it relies on circuit approximations, resulting in some effects not being correctly modeled, such as the decrease in voltage at the start of each clock phase or the initial voltage value due to the initial conditions, which is also correctly modeled in the numeric methodology.

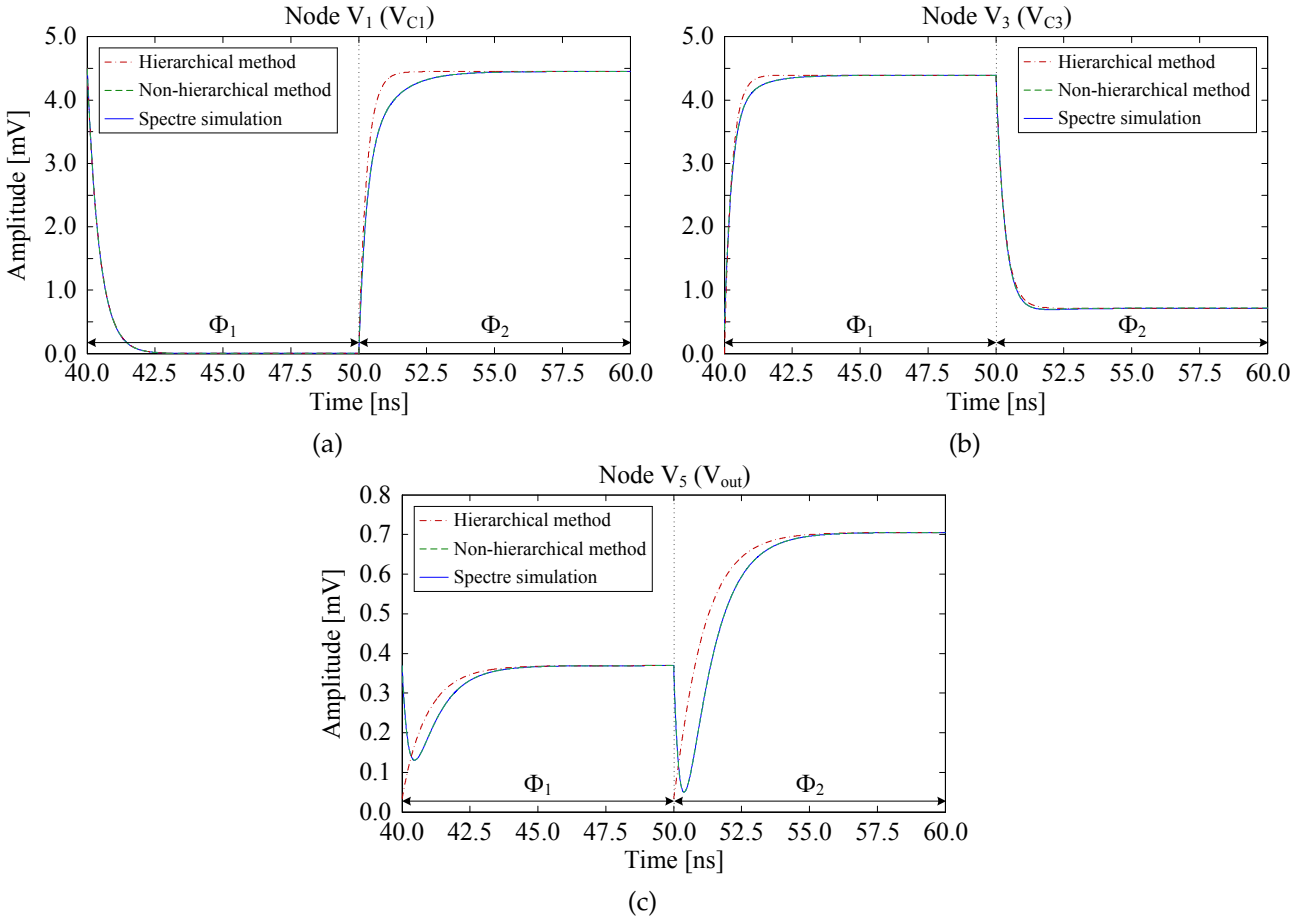


Figure 5.19: Time-domain comparison between the hierarchical and non-hierarchical method for nodes (a) V_1 (V_{C1}), (b) V_3 (V_{C3}), and (c) V_5 (V_{out}) during one sampling period.

To determine the reason why the closed-loop analysis method does not correctly model the non-ideal effects of the circuit components, the influence of the nodal initial conditions on the circuit's output voltage, using both methods presented in this chapter, was analyzed. This comparison is shown in Fig. 5.20.

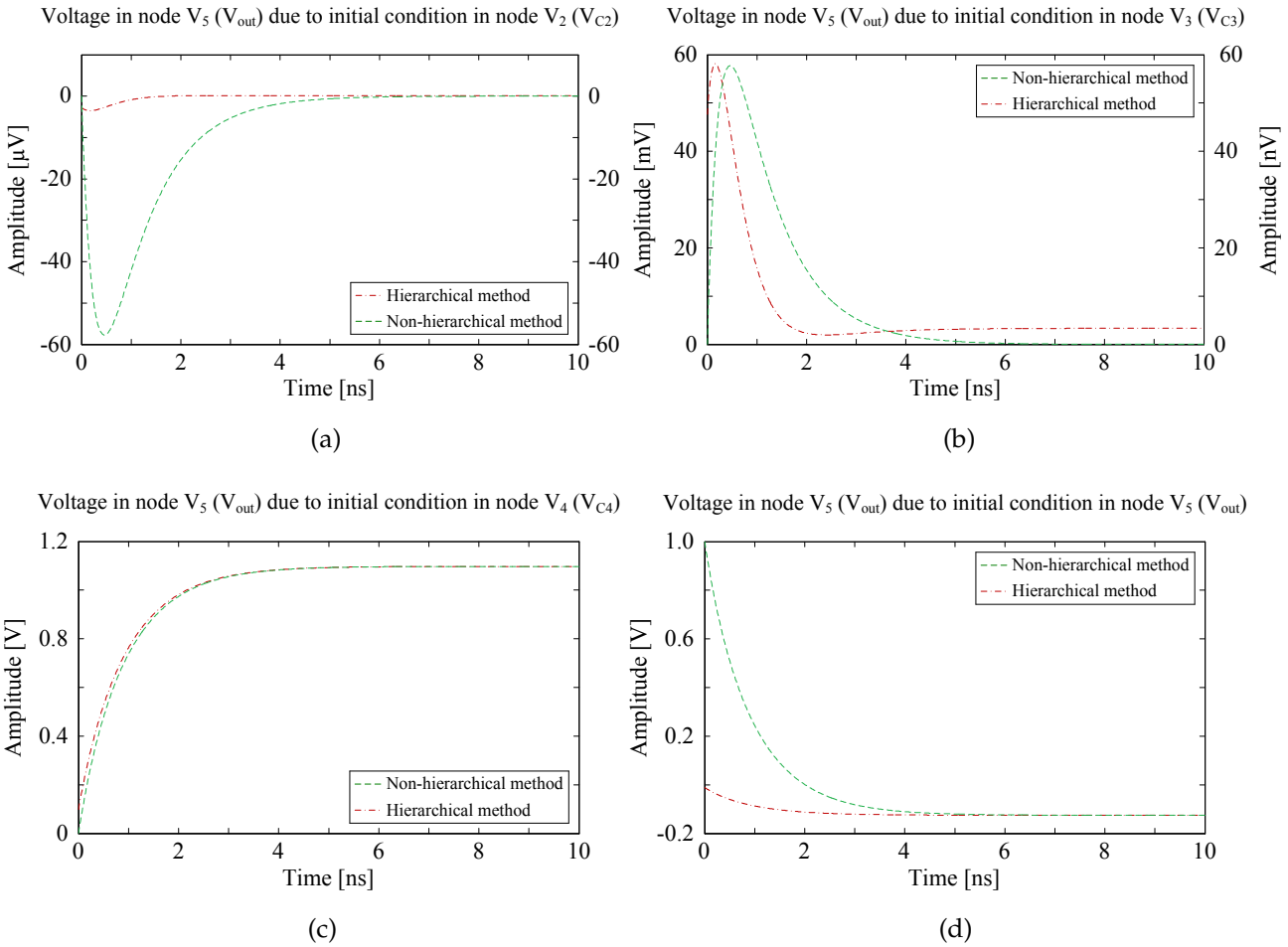


Figure 5.20: Effects of the initial conditions on the circuit's output voltage, during phase Φ_1, with the hierarchical and non-hierarchical method.[3]

The previous graphs shown that only the initial condition due to node V_4, which corresponds to the amplifier's input node, is correctly modeled. This is expected since the equations and the approximated circuits used to calculate the circuit's output voltage are based on the amplifier's input node and feedback paths that have a permanent path to this node. Since, in this filter architecture, there is no permanent path from the output to the input of the amplifier, through capacitor C_6, in either of the clock phases, the effect of nodes V_2, V_1 (during phase Φ_2) and V_3 (during phase Φ_1), i.e., the circuit components connected to these nodes, are not accounted for in the approximated model used, resulting in the errors in the starting voltage value and the decrease in voltage at the beginning of each phase.

[3] The influence of the initial condition from node V_1 (V_{C1}) is not shown in the figure since, during phase Φ_1, this node is not connected to the rest of the circuit.

5.3 Conclusions

In this chapter, two methods for the analysis of SC filters were described. In the first method the circuit is analyzed hierarchically, using circuit approximations to calculate the time-domain behavior of the amplifier's output node, using the closed-loop transfer function of the amplifier, and the behavior of the switches' resistance, using a first-order RC model. Since this method relies on circuit approximations, the effects of some of the components are not correctly modeled, leading to the need to design the circuit by excess (larger switches and more power dissipation in the amplifier) ensuring, with a higher degree of certainty, that the circuit is still capable of achieving charge equilibrium in every clock phase, even with the effects that are not modeled in the equations. This method is best suited for small circuits since the closed-loop transfer function can reach high orders, even for simple circuits, increasing computation effort necessary to decomposing this transfer function into first order functions and calculate their inverse Laplace transform to convert the output voltage into the time-domain.

In the second method the circuit is analyzed non-hierarchically, using a single system of equations to accurately describe the circuits' behavior, both in the time- and in the frequency-domain. This accuracy is possible since the system of equations includes all the effects of the transistors in the circuit, both in the amplifier and in the switches, by using the medium frequency small-signal model of the transistors. Unlike the first method, which is limited to compensate the non-ideal effects of the amplifier and of the switches, to obtain a solution as close as possible to the ideal design, being limited to designs that achieve charge equilibrium in every clock phase (complete settling), this method can accurately calculate the frequency response, even when charge equilibrium is not achieved by the end of a phase. Since the method relies on simple matrix computations it can be efficiently implemented in a computer.

CHAPTER 6

OPTIMIZATION METHODOLOGIES FOR THE DESIGN OF SWITCHED-CAPACITOR FILTERS

The transfer function of a filter circuit can change dramatically when the value of a single parameter changes. This makes determining a set of parameter values that produce the desired filtering transfer function and, at the same time, achieving other goals, such as small area and low sensitivity to component mismatch, a difficult task.

Traditionally, filter circuits are designed by first determining a prototype transfer function, using a mathematical approximation (e.g. Butterworth, Chebyshev, Elliptic), that meets the desired filter specifications. This prototype function is then used to calculate the component values that allow the implementation of the prototype transfer function. Since, in most cases, the number of circuit parameters is larger than the number of constraints imposed by the filter's prototype transfer function, additional constraints need to be used in order to have a unique solution for the circuit component values. However, this procedure is not efficient to find solutions with small area and low sensitivity.

An alternative is to use an optimization algorithm, such as genetic algorithms (GAs), to select the circuit parameters that make the filter's frequency response meet the desired specifications, skipping the need to choose a specific mathematical approximation and allowing the optimization procedure to directly find a set of circuit parameters that meet the desired specifications, while satisfying the other goals. In order to implement this optimization procedure, it is necessary to compute the transfer function of the circuit to compare it to the desired frequency response specifications and thus generate the part of the cost function of the GA related to the frequency response.

This procedure can be difficult because it needs to be time efficient, while taking into consideration all the non-ideal effects in the circuit. In the case of switched-capacitor (SC) circuits, this translates into considering the ON resistance and parasitic capacitances of the switches, the finite gain and bandwidth of the amplifiers, and the parasitic capacitances from the filter's capacitors.

The exact frequency response of a SC filter can be obtained from a transient simulation of the impulse response of the circuit and then calculate the fast Fourier transform (FFT) of the sampled output signal. However, this results in a large computation time, due to the length of the transient simulation necessary to get an accurate result, limiting the maximum size of the population that can used in the GA. As an alternative, the filter's frequency response can be estimated using equations, resulting in a low computation effort that allows the evaluation of larger populations in the GA in a reasonable amount of time.

In this chapter, the optimization procedure using GAs and two equation-based optimization methodologies for the design of SC filters are described. The first method, described in Section 6.2, uses a hierarchical approach (Section 5.1) to obtain the optimized filter design. Initially, the ideal filter, the amplifier circuit, and the switches are optimized (one at a time), using equations and a large population in the GA, increasing the odds of completely exploring the design space. Once the specifications are met, the size of the population is reduced and the final design of the complete filter is obtained using the more computation intensive, and more accurate, simulation-based cost function. The use of hybrid cost functions (equation-based followed by simulation-based), allows the exploration of larger design spaces and obtaining a design solution, with good accuracy, while still having a low computation time. The second method, described in Section 6.3, is similar to the previous one but uses a non-hierarchical approach (Section 5.2) to obtain the filter's frequency response, i.e., the optimized sizing of the capacitors and of the transistors (amplifier and switches) is obtained using a single system of equations, thus avoiding the multiple optimization steps of the hierarchical approach.

6.1 Optimization Procedure

The two analysis methods presented in Chapter 5 were implemented as software libraries and integrated into an existing circuit optimization software platform [74], [81], [82], based on the open-source circuit simulator Ngspice [83]. The optimization procedure, based on GAs, starts by generating a population of N randomly created chromosomes, bounded by the maximum and minimum values of each circuit component (capacitors, resistors, transistors, and voltage and current sources), defining the design space. Each chromosome is then evaluated, using hybrid cost functions (equation-based/simulation-based), and graded, using fitness functions. Once all chromosomes are evaluated and graded, their fitness is used to sort the population. The next generation is created based on the genetic material of the best chromosome and complemented with selection, crossover, and mutation operations among the remaining chromosomes. The general flow of the optimization procedure is shown in Fig. 6.1.

6.1.1 Chromosome Evaluation

In the optimization of SC filters, the following parameters are typically used in the cost function to evaluate the performance of the chromosomes.

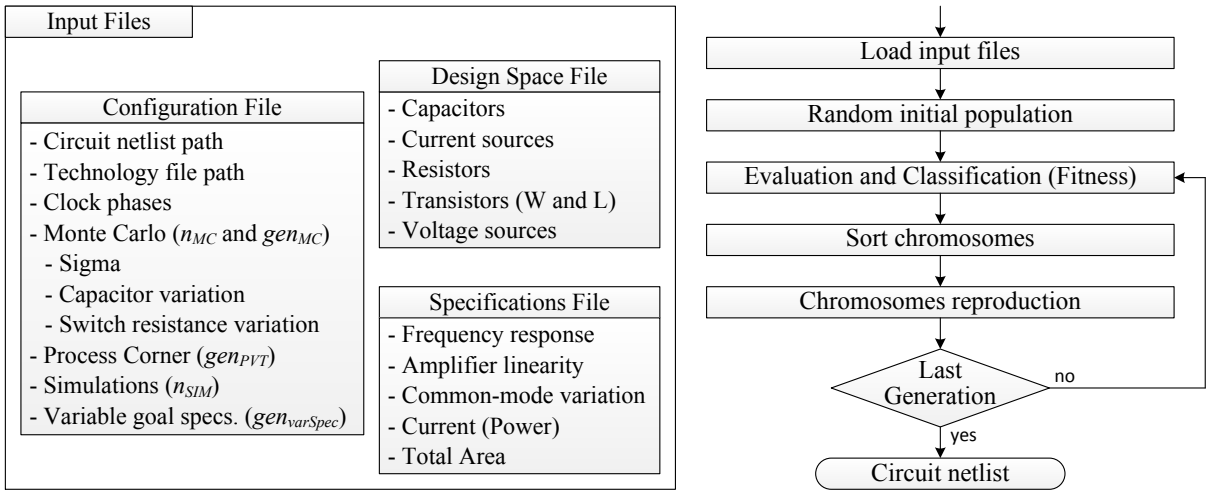


Figure 6.1: General design flow of the optimization procedure.

6.1.1.1 Amplifier linearity in the input voltage range

Considering the case of SC filters implemented using low gain amplifiers, the filter's frequency response, especially inside the passband, is going to be sensitive to variations in the amplifier's gain, increasing the distortion at the output of the filter. To minimize this effect, the amplifier's linearity within the desired input voltage range is used in the cost function. This parameter is obtained using DC simulations, sweeping the input common-mode voltage (V_{cmi}) from ground (V_{SS}) to the supply voltage (V_{DD}).

The indicators used for the evaluation of the amplifier's linearity are shown in Fig. 6.2, where V_{amp} represents the maximum signal amplitude at the input of the amplifier and where ΔG represents the gain variation in the input voltage range, which is the indicator that is going to be minimized in the optimization. To avoid the simulation of unnecessary points, the linearity is only evaluated in the input voltage range $[V_{cmi} - V_{amp}, V_{cmi} + V_{amp}]$ and for a small number of points ($n_{points} = 5$), keeping the evaluation time negligible.[1]

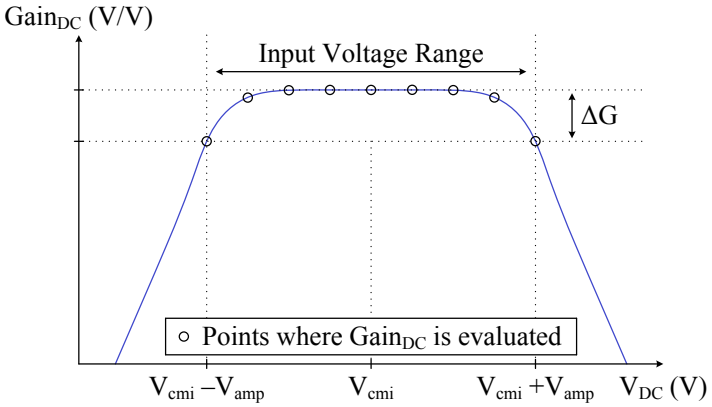


Figure 6.2: Indicators used to evaluate the fitness of the amplifier's linearity in the input voltage range.

[1] For amplifiers in differential configuration, a small offset ($\pm$ 10 μV) is used when evaluating the gain for the common-mode voltage to avoid the division by zero from $Gain_{DC} = (v_{op} - v_{on})/(v_{ip} - v_{in})$.

6.1.1.2 Amplifier gain

Depending on the optimization methodology used, it might be necessary to have the gain of the amplifier as a design specification. The gain can be obtained using the amplifier's transfer function $H(s)$, considering $s = 0$, and a DC simulation to obtain the parameters of the transistors (gm, gds, and gmb). Alternatively, the gain value can be obtained from the swept DC simulation used to evaluate the amplifier's linearity at the common-mode voltage ($V_{cmi} \pm 10\ \mu V$).

6.1.1.3 Amplifier settling time

To determine if the amplifier is fast enough to drive the output of the SC filter in each clock phase, the time it takes for the amplifier's output to remain within a given settling error after injecting a step input can be, depending on the optimization methodology, used in the chromosomes' cost function. The output is obtained from the inverse Laplace transform of the amplifier's closed-loop transfer function (5.3). The number of points where this function is evaluated depends on the desired accuracy for the settling time.

The indicators used in the evaluation of the settling time are shown in Fig. 6.3, where the settling error is given by $\varepsilon_{settling}$, the settling time by $t_{settling}$, and t_{ph} represents the length of the clock phase.

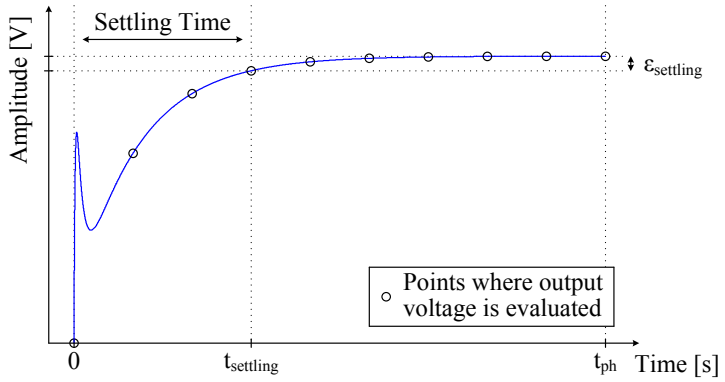


Figure 6.3: Indicators used to evaluate the fitness of the amplifier's settling time.

6.1.1.4 Amplifier output common-mode voltage

When SC filters are implemented using low gain amplifiers, it is not strictly necessary to use a common-mode feedback (CMFB) circuit to obtain a given common-mode voltage at the output. Instead, the output common-mode voltage, which is obtained from a DC simulation, can be used in the cost function.

6.1.1.5 Frequency response of the filter

To determine if the filter meets the desired specifications, the frequency response is evaluated at several key frequencies, as shown by the blue solid line in Fig. 6.4 and Fig. 6.5, and where the gray areas represent forbidden regions.

Considering the case of frequency responses of lowpass filters (Fig. 6.4), the gain inside the passband is controlled by indicators $F_{passlow}$ and F_{cutoff}, where the gain is evaluated at n linearly spaced frequencies between these two points to ensure that it is between A_{max1} and A_{max2}. To place the poles at the correct frequency (F_{cutoff}), indicator F_{stop} is used. At this frequency the gain should be below A_{min1}, forcing the poles to move from higher frequencies down to the desired cutoff frequency. In this example, where there are no zeros, indicators $F_s/2$ and A_{min2} are used to reinforce the poles placement. In architectures capable of implementing bilinear type transfer functions, these indicators are used to place the zeros of the transfer function at $F_s/2$.

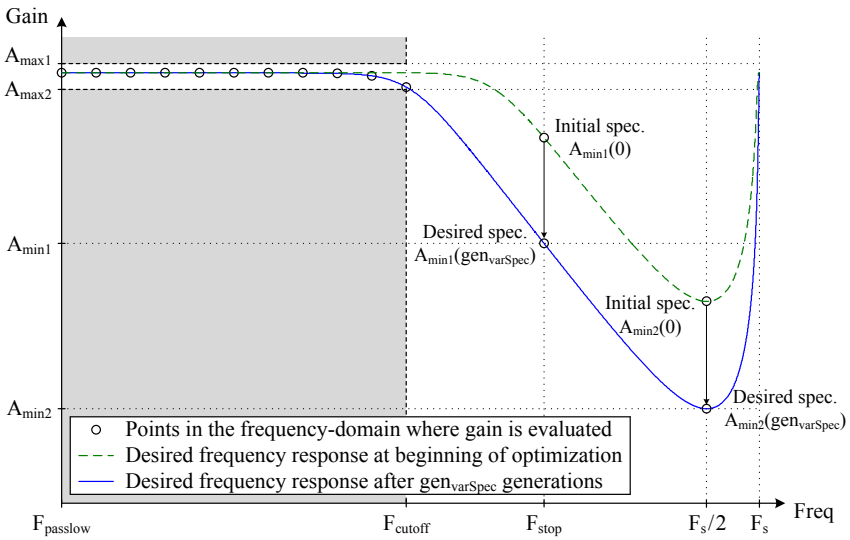


Figure 6.4: Indicators used to evaluate the fitness of the frequency response of lowpass SC filters.

In the case of bandpass filters (Fig. 6.5), the gain inside the passband is controlled in the same manner as in lowpass filters, where the passband frequency is controlled by indicators $F_{passlow}$ and $F_{passhigh}$ and the gain by indicators A_{max1} and A_{max2}. The filter's quality factor, i.e., the location of the poles, is controlled by indicators $F_{stoplow}$ and $F_{stophigh}$ where, at these frequencies, the gain should be below A_{min1} and A_{min2}, respectively. Indicator $F_s/2$ has the same function as in lowpass filters.

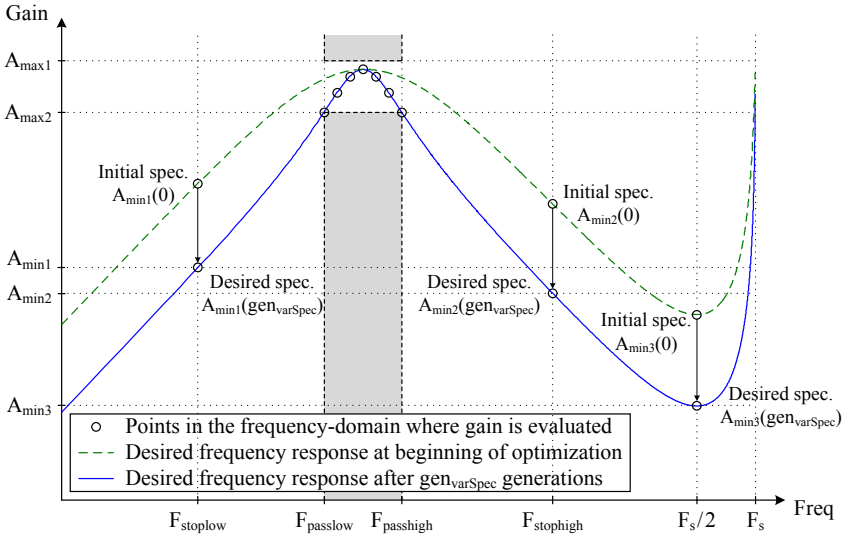


Figure 6.5: Indicators used to evaluate the fitness of the frequency response of bandpass SC filters.

6.1.1.6 Power consumption and area

To prioritize solutions with low power and small area, these two parameters can also used in the cost function during the optimization process.

6.1.2 Chromosome Grading (Fitness)

Once the parameters used in the cost function are obtained, the chromosome's fitness is calculated, using one of three different types of exponential-based equations (6.1), depending on the optimization action required for each specification (maximize, equalize, or minimize goal) [84]. Note that the chromosome's fitness is given by the product of the fitness of each specification.

$$\begin{aligned} &\text{Maximize parameter:} && f_{max} = 1 - e^{-weight^{-1} \times (achieved/desired)} \\ &\text{Equalize parameter:} && f_{equ} = \frac{2}{e^{-\gamma} + e^{\gamma}}, \quad \gamma = weight \times \frac{achieved - desired}{desired} \\ &\text{Minimize parameter:} && f_{min} = 1 - e^{-weight^{-1} \times (desired/achieved)} \end{aligned} \tag{6.1}$$

where, the variable *desired* represents the desired value for a given parameter (area, power, attenuation at a given frequency, etc.) and *achieved* the value obtained, either through simulation or by using equations, for that parameter. Variable *weight* controls the importance given to each parameter, i.e., to obtain the same fitness value when using a higher *weight* value, a given parameter needs to be closer to the desired value, as shown in Fig. 6.6.

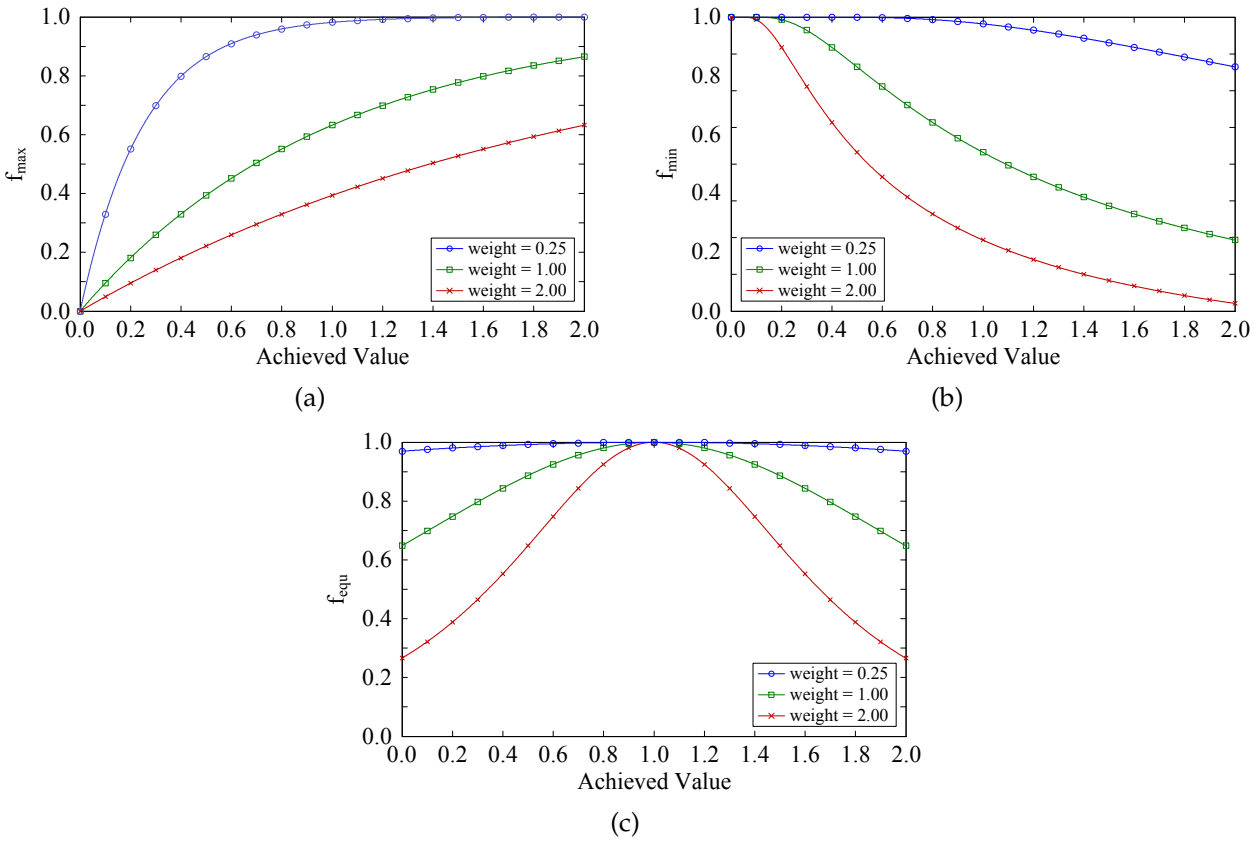


Figure 6.6: Fitness function (a) maximize, (b) minimize, and (c) equalize goal as a function of *achieved* for different *weight* values and *desired* = 1.0.

6.1.3 Types of Optimization

Since the problem of optimizing a filter circuit is very difficult and time consuming, the way the fitness is computed changes during the optimization procedure. The idea is to use different types of optimization to minimize the time it takes to converge to a solution within the desired specifications and that is robust against process, voltage supply, and temperature (PVT) corners and component mismatches.

6.1.3.1 Type I - Optimization using fixed goal specifications

This is the standard type of optimization used in GAs, where the desired values used in the fitness functions (6.1) are fixed throughout the optimization procedure. If the parameters being optimized have quite demanding specifications, and considering that the first population in the GA is random, the chances of finding a chromosome that produces results close to the desired specifications, during the first generations is very small, making it extremely difficult to find the global minimum in the design space. This can result in a slow convergence speed to the desired solution.

6.1.3.2 Type II - Optimization using varying goal specifications

This type optimization uses varying goal specifications in the more demanding parameters, where the desired value changes as a function of the current generation number, during a set number ($gen_{varSpec}$) of generations [5].

$$indicator(gen) = desired\_spec + (initial\_spec - desired\_spec)\, e^{-speed \times gen} \tag{6.2}$$

where, *initial_spec* and *desired_spec* are the initial and final values of the goal, respectively, *speed* is the speed at which the indicator converges to the final value (*desired_spec*), and *gen* is the current generation number. Note that the value of *speed* should be high enough to ensure that the indicator reaches the final goal value before $gen_{varSpec}$ generations have been evaluated.

The objective of the varying goal specifications is to facilitate the convergence process of the GA, by using more relaxed indicators at the beginning of the optimization (making it easier to find local minima) and, as the number of generations in the GA increase, the value of the indicators get closer to their ultimate goal.

It is important to notice that, in order to achieve the desired frequency response, the poles of the SC filter have to be located precisely in the $\mathcal{Z}$-plane. This means that the poles should have a certain frequency and quality factor. More stringent specifications correspond to higher quality factor values because the filter needs to have a rapid change in its attenuation from the passband to the stopband. By relaxing the specifications, the GA can first converge to a solution where the poles of the circuit are placed close to the desired frequency (represented by the dashed green line in Fig. 6.4 and Fig. 6.5) and then, as the specifications are tightened, the poles' quality factor can slowly increase to the correct value (represented by the solid blue line in Fig. 6.4 and Fig. 6.5). To ensure this slow increase, during this evaluation the fitness of indicators A_{min1}, A_{min2}, and A_{min3} are evaluated using the matching value fitness function (f_{equ}) instead of the maximize value function (f_{max}).

The effectiveness of using varying goal cost functions to increase the probability of converging to the best solution, in the smallest number of generations, is given in Section 6.2.2. The values chosen for $gen_{varSpec}$ and *speed* will depend on the circuit's complexity and on the size of the population used in the GA.

6.1.3.3 Type III and IV - PVT corner and mismatch (Monte Carlo) optimizations

To increase the chromosomes' robustness under different PVT corners and to Monte Carlo (MC) variations (mismatch errors), the transistors in the circuit are optimized against several corners and the filter's frequency response is optimized against mismatch errors in the capacitors. To ensure that the filter's time constants are much smaller than the phases length, i.e., that charge equilibrium is achieved before the end of each phase, random variations are also introduced in the switches ON resistance.

In the PVT optimization, each chromosome is evaluated under different PVT corners. This involves changing the devices' process (*SS*, *TT*, *FF*), the value of the voltage sources ($V_{DD} \pm 0.1 \times V_{DD}$), and the operating temperature before the evaluation of each corner. The performance of the chromosome is given by a combination between the fitness of the general chromosome ($TT/V_{DD} \pm 0/temp_{nom}$) and the lowest (worst-case) value of all the corner evaluations performed on the chromosome.

$$fitness_{PVT} = (1 - weight_{PVT}) \times fitness_{general} + weight_{PVT} \times fitness_{worst} \tag{6.3}$$

In the MC optimization, each chromosome is evaluated n_{MC} times, with random variations, corresponding to the expected mismatch errors of the capacitors and of the switches ON resistance [4]. The number of Monte Carlo evaluations per chromosome (n_{MC}) should be low (typically between 3 and 5), since this evaluation is performed during several generations, corresponding to many random variations tested. The performance of the chromosome is given by the worst fitness of the (n_{MC}) chromosomes.

6.1.4 Chromosome Reproduction

6.1.4.1 Selection Scheme

The selection scheme used to populate the new generation uses the elitism approach, copying the best chromosome to the new population without any change to the values of its genes. The remaining chromosomes are made by randomly selecting parents, using the Rank method, and applying the crossover and mutation operations to these chromosomes.

In the Rank method, the chromosomes are ordered based on their fitness value. The probability of these chromosomes being selected and used in the creation of the new population is proportional to their fitness value, i.e., chromosomes with a better fitness value have a higher probability of being selected to be used in the creation of the new population.

This selection scheme ensures that new generations are provided with the best genetic material from the previous generation, allowing the GA to converge to a good solution (local/global minimum), while still exploring other regions of the design space that may contain a better solution (global minimum).

6.1.4.2 Crossover Operator

After the population of the new generation, each chromosome, except for the elite, which remains unchanged, are combined to produce new genetic material, i.e., to introduce changes in the values of some of the genes.

The (one-point) crossover is performed by randomly selecting two chromosomes, based on their fitness. Afterwards, and for each gene (represented in binary format), the bits after a given cross point are exchanged between both chromosomes. The gene's cross points are randomly selected based on the GA current generation number.

At the beginning of the optimization, the cross point has a higher probability of exchanging more bits, introducing larger changes in the genes. At later generations, the cross point tends to exchange less bits since the GA should already be close to the global minimum, only requiring small changes in the values of the bits. An example of an one-point crossover is shown in Fig. 6.7.

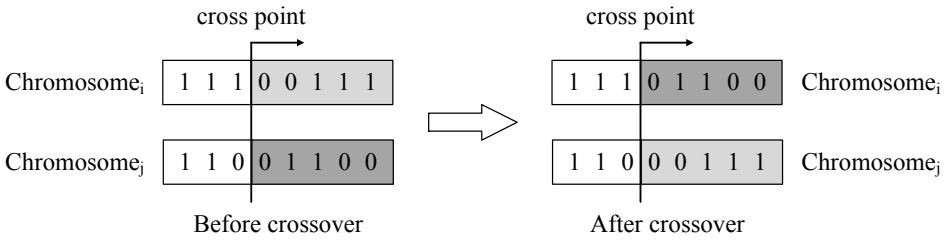


Figure 6.7: Example of the one-point crossover operator.

6.1.4.3 Mutation

The mutation operator is used to introduce diversity in the genetic material of a population. The number of chromosomes that are mutated is typically low, to avoid turning the GA into a random search. The chromosomes are selected randomly and each of its gene will suffer a mutation of a single bit. The bit to be mutated is randomly chosen based on the generation number.

At the beginning of the optimization, every bit in the gene can be mutated, to allow the full exploration of the design space. As the current generation number increases, the most significant bits will gradually begin to be left out of those that qualify to be mutated. This is done to prevent significant variations in the genes' value, as the GA approaches the optimum solution. An example of a mutation is shown in Fig. 6.8.

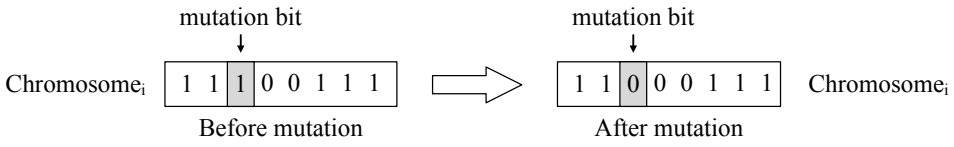


Figure 6.8: Example of the mutation operator.

6.2 Hierarchical Optimization Methodology

6.2.1 Method Description

The hierarchical optimization methodology, which uses the analysis method presented in Section 5.1, uses a multi-step approach to optimize SC filters. Initially, the filter is optimized from an ideal standpoint, using the equation that describes its transfer function. Once the desired filter specifications are met, the second step starts, where the software focuses on optimizing the amplifier circuit, at transistor level, in order to have enough gain and closed-loop bandwidth to replace the ideal gain value used in the top-level optimization. In the third step, the switches are optimized to have time constants that are small enough to charge the capacitors in every clock phase. Once the specifications are met, the values of the parasitic capacitances from the amplifier and from the switches are compensated into the ideal capacitor values obtained in the top-level optimization. Finally, the last step runs transient simulations of the impulse response to validate the equation-based design. The simplified diagram of this approach is shown in Fig. 6.9.

6.2.1.1 First Step: Top-Level Optimization

The first step is to optimize the SC filter, considering every component as ideal (capacitors, switches, and amplifier), using the filter's discrete-time transfer function to obtain the ideal frequency response, as described in Section 5.1.1.1. The filter's chromosome contains the values of the capacitors and the DC gain of amplifier.

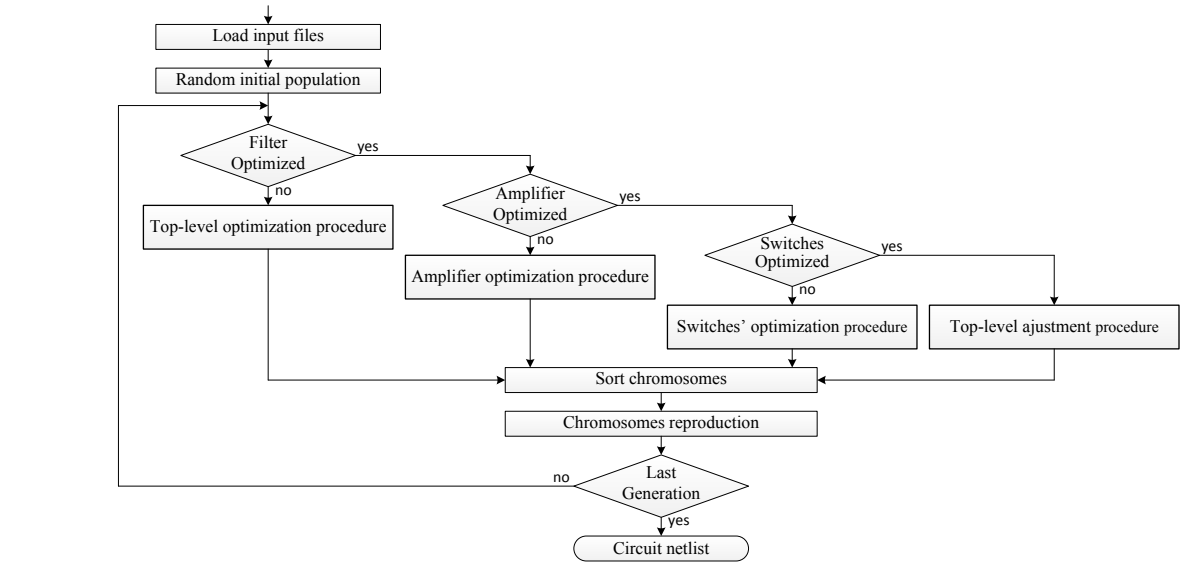
During this step, besides the optimization of the filter's frequency response, the software will also try to minimize the gain value necessary for the amplifier, since this will make the optimization of the amplifier circuit, in the second step, easier, as this value will be used as a design specification. The total capacitance value of the filter is also minimized to save area.

To minimize the amount of generations it takes for the GA to converge to the desired solution, this optimization step uses varying goal specifications (Type II) on the frequency response indicators (A_{min1}, A_{min2}, and A_{min3}), as described in Section 6.1.3.2, during, at least, $gen_{varSpec}$ generations and fixed goals on the remaining parameters. After this number of generations, and if there is at least one solution within the desired specifications, the software will switch to the MC optimization (Type IV), evaluating each chromosome n_{MC} times with random variations in the capacitors. The software will stay in this state during, at least, gen_{MC} generations, or until finding a solution that remains within the desired specifications, even with the random variations in the capacitors.

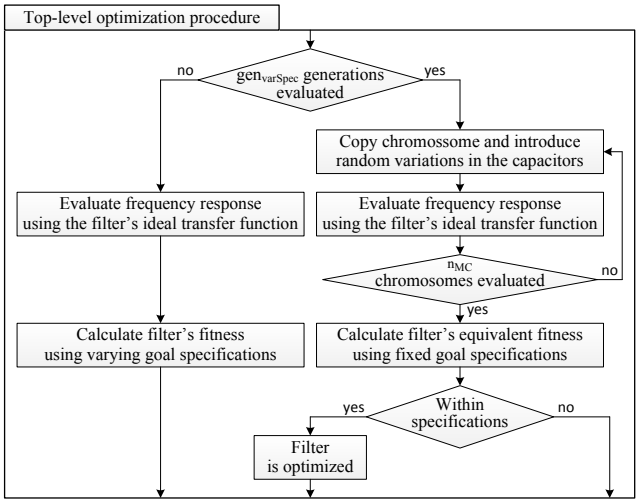
Considering a lowpass filter, the filter's cost function is given by (6.4) during the first $gen_{varSpec}$ generations and by (6.5) after these generations and during the MC optimization.

$$fitness_{filter} = \underbrace{f_{max}(A_n \,\epsilon\, [A_{max1}, A_{max2}]) \times f_{equ}(A_{min1}) \times f_{equ}(A_{min2})}_{\text{frequency response indicators}} \times f_{min}(G_{amp}) \times f_{min}(Area_{filter}) \quad (6.4)$$

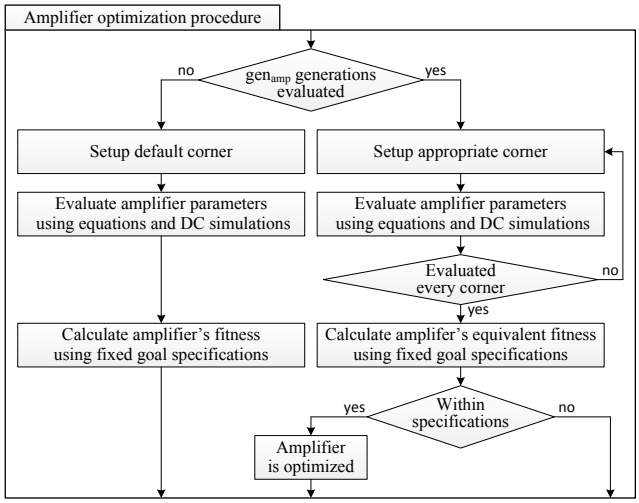
$$fitness_{filter} = \overbrace{f_{max}(A_n \,\epsilon\, [A_{max1}, A_{max2}]) \times f_{max}(A_{min1}) \times f_{max}(A_{min2})} \times f_{min}(G_{amp}) \times f_{min}(Area_{filter}) \quad (6.5)$$



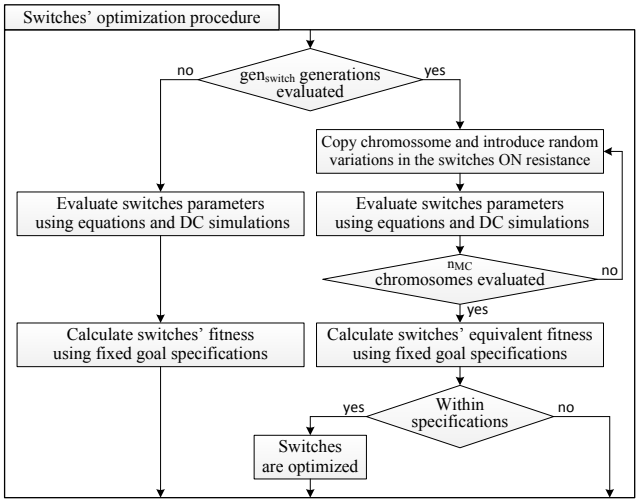
(a)



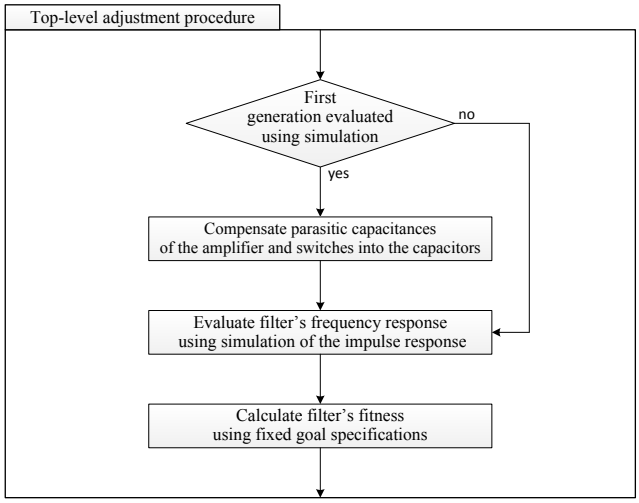
(b)



(c)



(d)



(e)

Figure 6.9: Flow of the hierarchical optimization methodology: (a) simplified diagram of the optimization procedure, and diagrams of the (b) ideal filter, (c) amplifier, (d) switches, and (e) complete filter evaluation and classification procedure.

6.2.1.2 Second Step: Amplifier Optimization

Once a robust solution is found for the filter's capacitors, the software changes to the optimization of the amplifier circuit, where the necessary gain value is obtained from the top-level optimization. The amplifier is optimized following the time-domain methodology described in Section 5.1.1.2. The amplifier's chromosome contains the values of the widths and lengths of the transistors, the bias current(s), and the input common-mode voltage.

In this step, the parameters that are going to be minimized in the cost function are the amplifier gain variation in the input voltage range, the amount of time it takes for the output of the amplifier to reach a given settling error (settling time), and the power consumption and area of the amplifier. The output common-mode voltage (V_{cmo}) is also used in the cost function, depending on the situation, it might be necessary to either minimize it or match it to a given value. The closed-loop gain (CLG) is the most important parameter used in the amplifier's cost function, since even small gain variations can significantly impact the filter's frequency response, especially inside the passband which, due to the hierarchical nature of this optimization method, would have to be compensated in the last step (simulation-based) or involve a second passage through the top-level optimization step.

The desired value for the CLG is obtained from the ideal closed-loop transfer function (5.17), considering $s = 0$ and the value of the ideal gain block (G_{amp}) as the value obtained from the top-level optimization. The actual parameter value is obtained from (5.2). Alternatively, the amplifier gain can be analyzed, using one of the methods described in Section 6.1.1.2, instead of the CLG.

During this step, the software uses fixed goal specifications (Type I) in each parameter of the cost function. This is done for, at least, gen_{amp} generations, or until the specifications are met. After this number of generations, and if there is at least one solution within the desired specifications, the software will switch to the PVT optimization (Type III), evaluating each chromosome under different PVT corners. The software will stay in this state during, at least, gen_{PVT} generations, or until finding a solution that remains within the desired specifications, even under different corners.

Before changing to the next optimization step, the values corresponding to the amplifier's input and input-to-output parasitic capacitances are stored, to be later compensated into the ideal capacitor values obtained in the top-level optimization. The amplifier's cost function is given by (6.6).

$$fitness_{amplifier} = f_{equ}(CLG) \times f_{min}(t_{settling}) \times f_{min}(\Delta G) \times f_{min}(Power) \times f_{min}(Area_{amp}) \times f_{min}(V_{cmo}) \quad (6.6)$$

6.2.1.3 Third Step: Switches' Optimization

In the third step, the switches' dimensions are optimized to occupy the minimum amount of area, while still having a time constant small enough to charge the capacitors in the phases' duration. Depending on the amplifier's influence on the settling of the nodes, the time constants can be calculated as described in Section 5.1.1.3, using a similar approach to the one used to optimize the amplifier circuit or, since this method is computationally intensive to use in every node, greatly increasing the time taken to obtain an optimized solution, first-order RC approximations can be used on the less sensitive nodes. The switches' chromosome contains the values of the widths and the length of the transistors.

During this step, the software uses fixed goal specifications (Type I) in each parameter of the cost function. This is done for, at least, gen_{switch} generations, or until the specifications are met. After this number of generations, and if there is at least one solution within the desired specifications, the software will switch to the MC optimization (Type IV), evaluating each chromosome n_{MC} times with random variations in the switches ON resistance. The software will stay in this state during, at least, gen_{MC} generations, or until finding a solution that remains within the desired specifications, even with random variations.

Before changing to the next optimization step, the values corresponding to the switches' parasitic capacitances ($C_{dd_{on/off}}$ and $C_{ss_{on/off}}$) are stored, to be later compensated into the ideal capacitor values obtained in the top-level optimization. The switches' cost function is given by (6.7), where n_{switch} represents the number of switches in the SC filter.

$$fitness_{switches} = f_{min}(Area_{switches}) \times \prod_{i=1}^{n_{switch}} f_{min}(\tau_{switch_i}) \tag{6.7}$$

6.2.1.4 Fourth Step: Top-Level Adjustment

The last optimization step is used to validate the equation-based design through electrical simulations and to fine-tune the capacitor values. The filter's transfer function is now obtained by running a transient simulation of the impulse response of the complete filter circuit. This procedure is very computational intensive and takes some time, making it difficult to use a large population or a large number of generations in this step. Therefore, the capacitors' design space is restricted to be around the values of the best chromosome obtained previously and the size of the population (n_{SIM}) is very small (typically between 5 and 10 chromosomes).

In this step, the capacitor values are adjusted to take into consideration the parasitic capacitances previously calculated. While the calculation of the amplifier's parasitic capacitances is relatively straightforward, since their value is constant for a fixed voltage, the value of the switches' parasitic capacitances depend on the state of the switch (ON/OFF) and can have a variation of over five times between the two states, making the compensation process more complex.

To overcome this problem, the chromosomes of the first generation are all identical apart from a random compensation factor (C_{factor}), which varies between 0 and 5 (6.8), while the amplifier's parasitic capacitances are simply subtracted to the ideal capacitor values. From this initial population, it is possible to converge to an optimized solution very rapidly, i.e., within a few generations.

$$C_{compensated} = C_{general} - C_{factor} \times C_{par} \qquad 0 < C_{factor} < 5 \tag{6.8}$$

where, $C_{general}$ and $C_{compensated}$ represent, respectively, the capacitor value before and after the compensation of the switches' parasitic capacitances, C_{factor} the random compensation factor used to estimate the influence of the parasitic capacitances, and C_{par} the approximated value for the parasitic capacitances, which is given by the average between the ON and OFF values. The cost function used in the top-level adjustment step is given by (6.9).

$$fitness_{complete\_filter} = \underbrace{f_{max}(A_n \,\epsilon\, [A_{max1}, A_{max2}]) \times f_{max}(A_{min1}) \times f_{max}(A_{min2})}_{\text{frequency response indicators}} \times f_{min}(Area_{filter}) \tag{6.9}$$

6.2.2 Example I: Second-Order Lowpass SC Filter

In this section, the second-order lowpass SC filter described in Section 5.1.2, reproduced in Fig. 6.10 for convenience, is used to demonstrate the optimization method presented in Section 6.2.1 and to show the advantage of using varying goal specifications to converge to the desired solution in a smaller number of generations [77]. For simplicity purposes the circuit is shown in single-ended configuration. In differential configuration capacitors $C_{1,2,3,4}$ are implemented with two capacitors in anti-parallel.

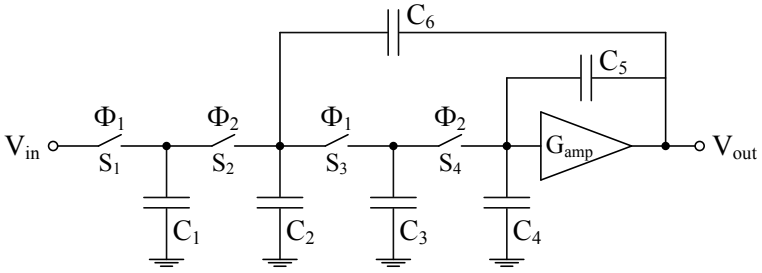


Figure 6.10: Second-order lowpass SC filter.

The circuit was designed in a standard 130 nm CMOS technology and the capacitors were implemented using MIM capacitors. Although the optimization software is integrated with the open-source simulator Ngspice, the chromosomes obtained from optimization are also simulated in Spectre to validate the results.

The SC filter's frequency response was optimized using with the design specifications shown in Table 6.1. Besides the optimization of the frequency response, the area occupied by the capacitors is also minimized. The chromosome and the lower and higher bounds of the design space used in this step are shown in Table 6.2.

Table 6.1: Specifications used to optimize the frequency response of the lowpass SC filter.

$F_{passlow}$ (F_{pl})	F_{cutoff} (F_c)	F_{stop}	$F_s/2$
10 kHz	1 MHz	6 MHz	50 MHz
0.01 dB $\geq$ $[F_{pl}, F_c]$ $\geq$ -3.01 dB		$\leq$ -30.00 dB	$\leq$ -59.00 dB

Table 6.2: Chromosomes and design space bounds used by the GA in the top-level optimization.

Design variables	C_1 (pF)	C_2 (pF)	C_3 (pF)	C_4 (pF)	C_5 (pF)	C_6 (pF)	G_{amp} (V/V)
Lower bound	0.05	0.05	0.05	0.05	0.05	0.05	0.75
Higher bound	10.00	10.00	10.00	10.00	10.00	10.00	1.00

To evaluate the performance of fixed (Type I) and varying goal (Type II) specifications on the optimization speed of the filter's frequency response, 50 top-level optimizations were performed under the same conditions, i.e., same design space and same number of chromosomes in each population (2500), using fixed and varying goal specifications, during the first 30 generations, in indicators A_{min1} and A_{min2}. Results show that, on average, the software takes 96 generations to find a solution within the desired specifications when using fixed goal specifications and only 58 generations when using

varying goal specifications during the first 30 generations which, in this example, corresponds to a 40 % improvement in the convergence speed of the GA in the top-level optimization step. An example of a design obtained after 61 generations is shown in Table 6.3.

Table 6.3: Lowpass SC filter design obtained after the top-level optimization.

C_1 (fF)	C_2 (fF)	C_3 (fF)	C_4 (fF)	C_5 (fF)	C_6 (fF)	G_{amp} (V/V)
54.5	99.3	50.0	474.0	131.6	4530.8	1.0

In the second optimization step, since the amplifier circuit needs to have a linear gain in the input voltage range, the amplifier is now implemented using the topology shown in Fig. 6.11.

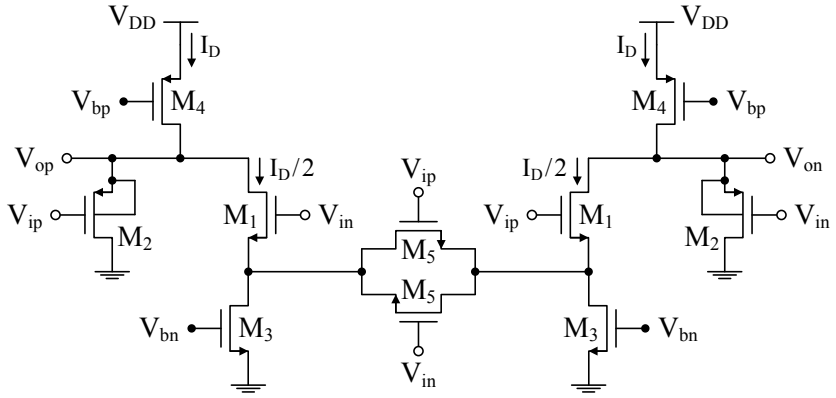


Figure 6.11: Differential voltage-combiner amplifier with source degeneration.

The amplifier's closed-loop transfer function was calculated using the same steps described in Section 5.1.2.2, replacing the amplifier block with the circuit from Fig. 6.11. Note that, since the amplifier now has an internal node, the approximated value for the input parasitic capacitance is now calculated in a different manner. In this case the compensated values are given by (6.10).

$$\begin{aligned} C_{4a} &= C_4 - C_{gb_1} - (1+G_1)C_{gd_1} - (1+G_2)C_{gs_1} - C_{gd_2} - C_{gg_5} \\ C_{5a} &= C_5 - C_{gb_2} - C_{gs_2} \end{aligned} \tag{6.10}$$

where, G_1 and G_2 represent, respectively, the voltage gain between the gate and the drain and the voltage gain between the gate and the source of transistor M_1.

The amplifier was optimized using the design specifications shown in Table 6.4, where the DC gain parameter is the value obtained from the top-level optimization. The chromosome and the lower and higher bounds of the design space used in this step are shown in Table 6.5.

Table 6.4: Amplifier specifications and simulation results of the best chromosome.

	DC gain	Settling time Φ_1 (%)	Settling time Φ_2 (%)	Power (µW)
Specifications	1.00	< 75.0	< 75.0	minimize
Optimization solution	1.00	59.9	65.1	261.4
Spectre simulation	1.00	55.2	65.2	261.4

Table 6.5: Chromosomes and design space bounds used by the GA in the amplifier optimization.

Design variables	W_1 (μm)	L_1 (μm)	W_2 (μm)	L_2 (μm)	W_3 (μm)	L_3 (μm)	W_4 (μm)	L_4 (μm)	W_5 (μm)	L_5 (μm)	I_D (nA)
Lower bound	0.16	0.12	0.16	0.12	2.00	0.12	2.00	0.12	2.00	0.12	70.00
Higher bound	100.00	1.00	100.00	1.00	100.00	1.00	100.00	1.00	100.00	1.00	150.00

During this step, each generation was composed of 2500 chromosomes and it took 10 generations, using fixed goal specifications, to obtain the design solution shown in Table 6.6. Afterwards, the values of the amplifier's parasitic capacitances are calculated, based on a DC operating point (OP) simulation, and compensated into capacitors C_4 and C_5 (represented as C_{4c} and C_{5c} in Table 6.6).

Table 6.6: Amplifier design obtained and compensated capacitances after the amplifier optimization.

W_1 (μm)	W_2 (μm)	W_3 (μm)	W_4 (μm)	W_5 (μm)	V_{DD} (V)	C_{4a} (fF)
0.37	62.30	39.20	90.50	0.38	1.20	1874.7
L_1 (μm)	L_2 (μm)	L_3 (μm)	L_4 (μm)	L_5 (μm)	V_{cmi} (V)	C_{5a} (fF)
0.52	0.24	0.35	0.45	0.22	0.45	27.0

In the third step, the filter's switches were optimized to minimize area and to have time constants small enough to charge the capacitors in less than the phases length. This was done using first order RC approximations, as described in Section 5.1.2.3. The chromosome and the lower and higher bounds of the design space used in this step are shown in Table 6.7.

Table 6.7: Chromosomes and design space bounds used by the GA in the switches optimization.

Design variables	W_{s_1} (μm)	W_{s_2} (μm)	W_{s_3} (μm)	W_{s_4} (μm)	L_s (μm)
Lower bound	0.16	0.16	0.16	0.16	0.12
Higher bound	1.00	1.00	1.00	1.00	0.12

The length chosen for the switches' transistors was fixed at the minimum value allowed by the technology. Since the SC filter is sensitive to the switches' parasitic capacitances, increasing the transistor's length would increase both resistance and parasitic capacitances of the switches. During this step, each generation was composed of 2500 chromosomes and it took 15 generations, using fixed goal specifications, to obtain the design solution shown in Table 6.8.

Table 6.8: Switch design obtained after the switches' optimization.

W_{s_1} (μm)	W_{s_2} (μm)	W_{s_3} (μm)	W_{s_4} (μm)	L_s (μm)
0.24	0.30	0.24	0.24	0.12

In the last step, the equation-based design is validated using transient simulations of the impulse response. Since the equations used have good accuracy, there is already a general idea where the global minimum is located in the design space. For this reason, the capacitors' design space is restricted to be around the solution obtained from the top-level optimization step (after compensating for the effects of the amplifier's parasitic capacitances). The chromosome and the lower and higher bounds of the design space used in this step are shown in Table 6.9.

Table 6.9: Chromosomes and design space bounds used by the GA in each optimization step.

Design variables	C_{1c} (pF)	C_{2c} (pF)	C_{3c} (pF)	C_{4c} (pF)	C_{5c} (pF)	C_{6c} (pF)
Lower bound	$0.95 \times C_1$	$0.95 \times C_2$	$0.95 \times C_3$	$0.95 \times C_{4a}$	$0.95 \times C_{5a}$	$0.95 \times C_6$
Higher bound	$1.05 \times C_1$	$1.05 \times C_2$	$1.05 \times C_3$	$1.05 \times C_{4a}$	$1.05 \times C_{5a}$	$1.05 \times C_6$

In this step, each generation was composed of 10 chromosomes and it took 12 generations, using fixed goal specifications, to obtain the final capacitor design, which is shown in Table 6.10. Although a solution, satisfying all the specifications, was found after a single generation in the top-level adjustment, using equation (6.8), the optimization process continued for another 11 generations to try to find a better solution than the one obtained after the first generation.

Table 6.10: Lowpass SC filter design obtained after the top-level adjustment.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)
53.2	97.3	47.8	459.3	27.7	4754.5

Since this method performs the optimization of the SC filter using the filter's ideal transfer function, relying on subsequent steps to replace the ideal blocks with real circuits with similar performance (amplifier and switches), while compensating for the non-ideal effects of the real blocks, the frequency response of the lowpass filter was evaluated after each step to show the correct compensation of the parasitic capacitances after each optimizations step. These results are shown in Table 6.11.

Table 6.11: Specifications and simulation results of the lowpass SC filter after each optimization step.[2]

	$F_{passlow}$	F_{cutoff}	F_{stop}	$F_s/2$
	10 kHz	1 MHz	6 MHz	50 MHz
(A)	$\leq$ 0.01 dB	$\geq$ -3.01 dB	$\leq$ -30.0 dB	$\leq$ -59.0 dB
(B)	0.00 dB	-2.95 dB	-30.78 dB	-59.87 dB
(C)	0.01 dB	-2.94 dB	-30.78 dB	-59.87 dB
(D)	0.00 dB	-2.91 dB	-30.73 dB	-59.89 dB
(E)	0.00 dB	-3.00 dB	-31.47 dB	-60.63 dB
(F)	0.00 dB	-3.03 dB	-31.50 dB	-60.61 dB

The results presented in the previous table show a good degree of accuracy between Spectre and the open source simulator Ngspice. When comparing rows (B) and (C), it can be concluded that the approximation used in equation (6.10), for the compensation of the amplifier's parasitic capacitances, is very accurate.

[2] (A) Specifications; (B) Equations/Spectre using ideal components; (C) Spectre - Real amplifier and ideal switches; (D) Ngspice - Real circuit before parasitic compensation; (E) Ngspice - Real circuit after (fine-tuning) optimization complete; (F) Spectre - Real circuit after (fine-tuning) optimization complete.

6.2.3 Example II: Second-Order Bilinear Bandpass SC Filter

In this section, a second-order bilinear bandpass SC filter, shown in Fig. 6.12, is used to demonstrate the MC optimization (Type IV). The filter architecture is based on the continuous-time bandpass Sallen-Key topology [35], replacing the resistors with parallel SC branches. Capacitor C_4 is used to introduce an additional zero in the filter's transfer function, allowing the implementation of bilinear transform based discrete filters. Capacitor C_9 allows additional charge to move between the output and the input of the amplifier, reducing the ratio between the largest and the smallest capacitor [4].

Note that, the filter was implemented in differential configuration. While, in this configuration, switches S_4 and S_6 discharge to the common-mode voltage, setting the input common-mode voltage of the amplifier (V_{cmi}), since the amplifier is implemented without a CMFB circuit, switch S_{10} connects to the other side of the differential circuit, due to the output common-mode voltage of the amplifier being different than V_{cmi}. Using this approach, during phase Φ_2, capacitor C_9 is connected to the correct reference voltage, independently of the amplifier's output common-mode voltage.

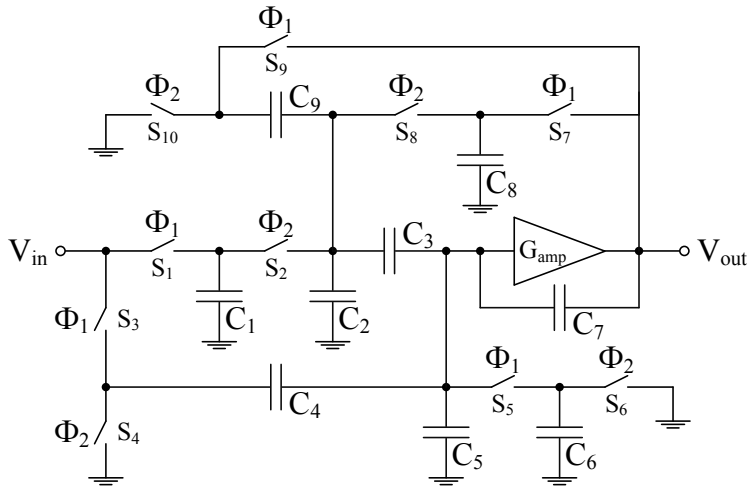


Figure 6.12: Second-order bilinear bandpass SC filter.

The circuit was designed in a standard 130 nm CMOS technology and the capacitors were implemented using MIM capacitors. The chromosome used in each optimization step is shown in Table 6.12.

Table 6.12: Chromosome format used by the GA in each optimization step.

Top-level optimization	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	G_{amp}	
Amplifier optimization	W_1	L_1	W_2	L_2	W_3	L_3	W_4	L_4	W_5	L_5	I_D
Switches' optimization	W_{s_1}	W_{s_2}	W_{s_3}	W_{s_4}	W_{s_5}	W_{s_6}	W_{s_7}	W_{s_8}	W_{s_9}	$W_{s_{10}}$	L_s
Top-level adjustment	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9		

The SC filter's frequency response was optimized using with the design specifications shown in Table 6.13. Besides the optimization of the frequency response, the area occupied by the capacitors and the direct current (DC) gain of the amplifier are also minimized during optimization.

Table 6.13: Specifications used to optimize the frequency response of the bilinear bandpass SC filter.

$F_{stoplow}$	$F_{passlow}$ (F_{pl})	F_c	$F_{passhigh}$ (F_{ph})	$F_{stophigh}$	$F_s/2$
0.2 MHz	7.7 MHz	10.0 MHz	12.3 MHz	20.0 MHz	50.0 MHz
$\leq$ -38.60 dB	0.01 dB $\geq$ $[F_{pl}, F_{ph}]$ $\geq$ -3.01 dB			$\leq$ -10.50 dB	$\leq$ -45.00 dB

In the first step, the filter was initially optimized, using varying goal (Type II) specifications, until reaching the desired frequency response specifications. Once the specifications were met, the software changes to the MC optimization (Type IV), evaluating each chromosome five times with random variations in the capacitors, corresponding to a Gaussian variable with a 3σ of 0.5 %.

The frequency response of the best chromosome before and after the mismatch optimization (for 10 different cases of random variations) is shown in Fig. 6.13 and Fig. 6.14 and the chromosomes are shown in Table 6.14. Before the mismatch optimization there was a 34 % probability of failing the passband specifications, while only 0.7 % after the mismatch optimization. The frequency response results show that the software not only chooses chromosomes that fit the desired specifications with random variations but also the one that has less variability with these variations.

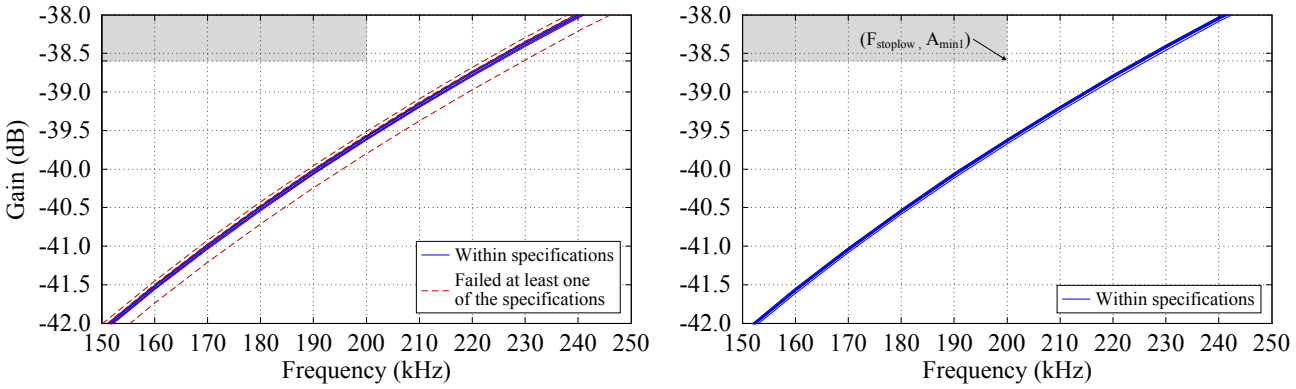


Figure 6.13: Frequency response of the best chromosome around the stopband frequency (a) before and (b) after the capacitor mismatch optimization.

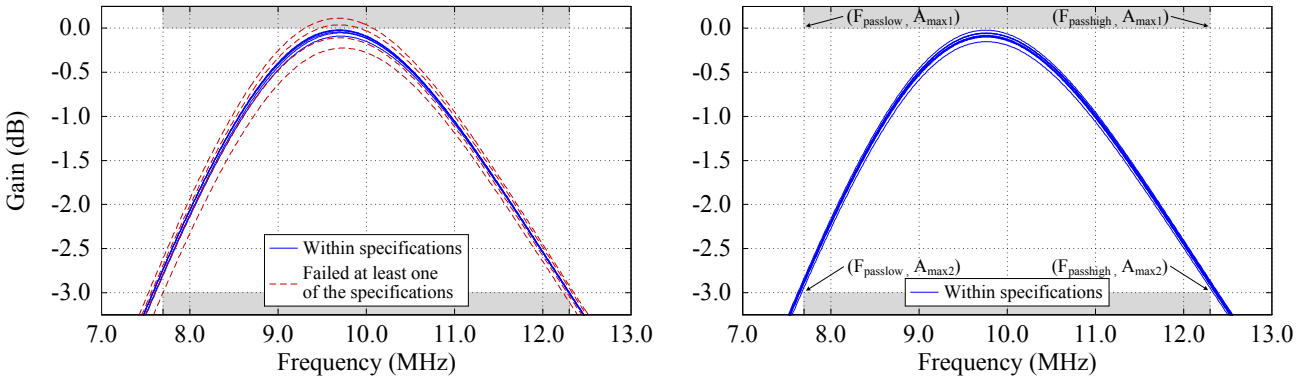


Figure 6.14: Frequency response of the best chromosome around the passband frequency (a) before and (b) after the capacitor mismatch optimization.

Table 6.14: Bandpass SC filter design obtained (A) before and (B) after mismatch optimization in the top-level optimization step.

	C_1 (fF)	C_2 (fF)	C_3 (fF)	C_4 (fF)	C_5 (fF)	C_6 (fF)	C_7 (fF)	C_8 (fF)	C_9 (fF)	G_{amp}
(A)	220.8	104.4	1869.0	107.2	125.0	220.8	348.5	778.2	2228.9	1.44
(B)	222.5	100.0	1639.4	100.2	126.9	224.5	331.6	760.9	2189.9	1.45

In the second optimization step, the amplifier's closed-loop transfer function was calculated in the same manner as in the previous example, considering the signal-flow diagram shown in Fig. 6.15 for clock phase Φ_1, where the filter's impedance seen by the output of the amplifier, is given by (6.11).

$$Z_{22}^{\Phi_1} = \left[\left(\frac{1}{s\,C_2} \parallel \frac{1}{s\,C_3}\right) + \frac{1}{s\,C_9} + R_9\right] \parallel \left[\frac{1}{s\,C_8} + R_7\right] \parallel \left[\frac{1}{s\,C_7}\right] \tag{6.11}$$

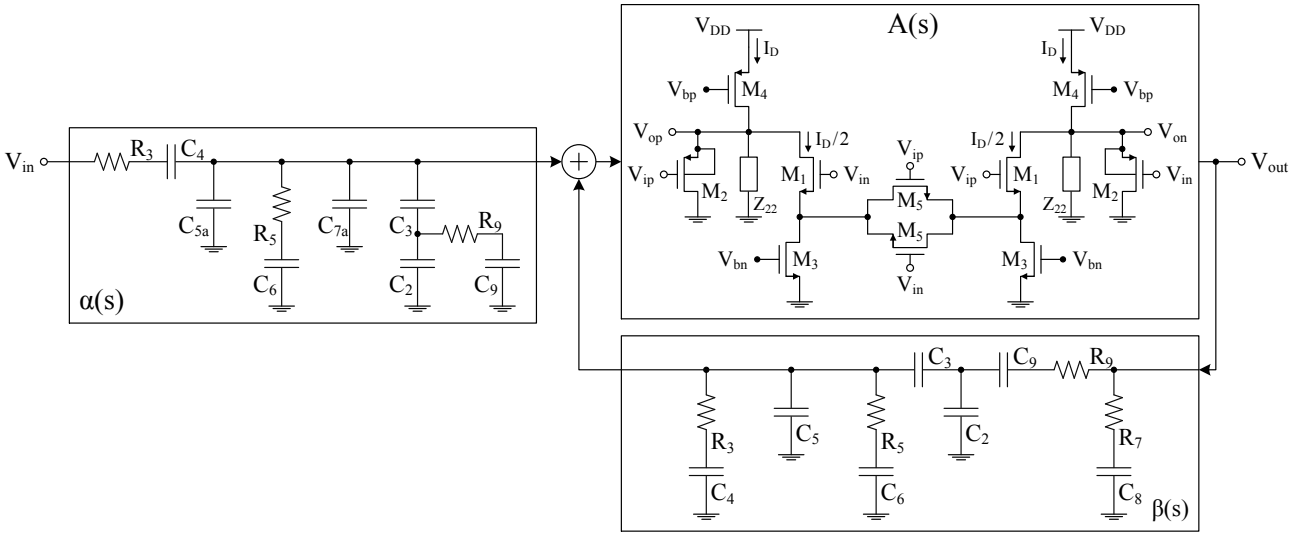


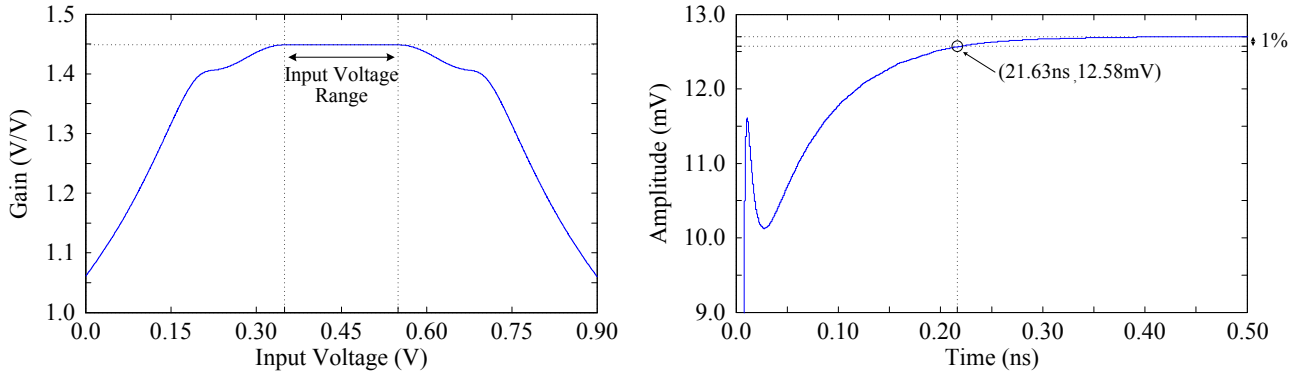
Figure 6.15: Closed-loop diagram of the bandpass SC filter during phase Φ_1.

The amplifier was optimized using the design specifications shown in Table 6.15, where the amplifier's linearity was optimized for a differential input signal with a maximum amplitude of 200 mV and the final value of the step response (SR_{end}) is calculated using the DC gain parameter obtained from the top-level optimization. Considering the complete filter circuit, and that the amplifier's DC gain $G_{amp} = 1.45$, the filter's input voltage range where the amplifier produces a linear gain is equal to $\approx 300\ mV$.

Table 6.15: Amplifier specifications and simulation results of the best chromosome.

	Linearity (mV/V)	SR_{end} (mV)	Settling time Φ_1 (%)	Power (µW)
Specifications	≤ 1.00	= 12.71	≤ 50.0	minimize
Optimization solution	0.39	12.71	45.9	976.8
Spectre simulation	0.41	12.70	43.3	976.8

The amplifier's DC gain and the filter's step response, obtained from a Spectre simulation, are shown in Fig. 6.16.



(a) (b)

Figure 6.16: Simulation results of the (a) amplifier's DC gain and the (b) filter's step response of the best chromosome obtained from the amplifier optimization step.

The chromosome obtained after this step is shown in Table 6.16, where the amplifier's input and input-to-output parasitic capacitances were compensated using (6.12).

$$\begin{aligned} C_{5a} &= C_5 - C_{gb_1} - (1+G_1)C_{gd_1} - (1+G_2)C_{gs_1} - C_{gd_2} - C_{gg_5} \\ C_{7a} &= C_7 - C_{gb_2} - C_{gs_2} \end{aligned} \tag{6.12}$$

where, G_1 and G_2 represent, respectively, the voltage gain between the gate and the drain and the voltage gain between the gate and the source of transistor M_1.

Table 6.16: Amplifier design obtained and compensated capacitances after the amplifier optimization.

W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	V_{DD} (V)	C_{5a} (fF)
12.10	83.20	253.60	41.40	82.30	1.20	250.8
L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	V_{cmi} (V)	C_{7a} (fF)
0.23	0.16	0.82	0.68	0.23	0.45	204.8

In this example, the filter's switches were designed with fixed values ($W_s = 5$ µm and $L_s = 0.13$ µm) to increase the influence of the switches' parasitic capacitances on the filter and to verify if, in the last optimization step, the software is still capable of finding a solution within the desired specifications using equation (6.8) on the first generation. Results shown that it was still possible to find a solution during the first generation using 10 chromosomes. The final capacitor design, shown in Table 6.17, was obtained after 10 generations of fine-tuning the capacitor values.

Table 6.17: Bandpass SC filter design obtained after the top-level adjustment.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)	C_{7c} (fF)	C_{8c} (fF)	C_{9c} (fF)
210.7	91.2	1631.6	97.9	62.8	219.6	204.8	733.9	2249.9

The filter's gain at the indicators' frequencies, after each optimization step, are shown in Table 6.18. Results show a good degree of accuracy, apart from a small gain difference at half the clock frequency for rows (E) and (F). Since the zeros of the transfer function are placed on top of the unit circle ($z = -1$), creating the effect of a notch with high attenuation, it is expected that, due to small differences between simulators and the way they calculate the effect of the parasitic capacitances, the zeros can move slightly away from the unit circle, causing substantial differences in the gain value.

Table 6.18: Specifications and simulation results of the bandpass SC filter after each optimization step.[3]

	$F_{stoplow}$	$F_{passlow}$	F_c	$F_{passhigh}$	$F_{stophigh}$	$F_s/2$
	0.2 MHz	7.7 MHz	10.0 MHz	12.3 MHz	20.0 MHz	50.0 MHz
(A)	$\leq$ -38.60 dB	$\geq$ -3.01 dB	$\leq$ 0.01 dB	$\geq$ -3.01 dB	$\leq$ -10.50 dB	$\leq$ -45.00 dB
(B)	-39.62 dB	-2.88 dB	-0.23 dB	-3.01 dB	-11.37 dB	-47.11 dB
(C)	-39.62 dB	-2.89 dB	-0.28 dB	-3.07 dB	-11.40 dB	-47.16 dB
(D)	-39.65 dB	-2.99 dB	-0.40 dB	-3.07 dB	-11.36 dB	-46.11 dB
(E)	-39.74 dB	-2.99 dB	-0.12 dB	-2.96 dB	-11.39 dB	-47.42 dB
(F)	-39.45 dB	-2.80 dB	-0.13 dB	-2.90 dB	-11.47 dB	-41.84 dB

The frequency response of the best chromosome once the optimization process is complete is shown in Fig. 6.17.

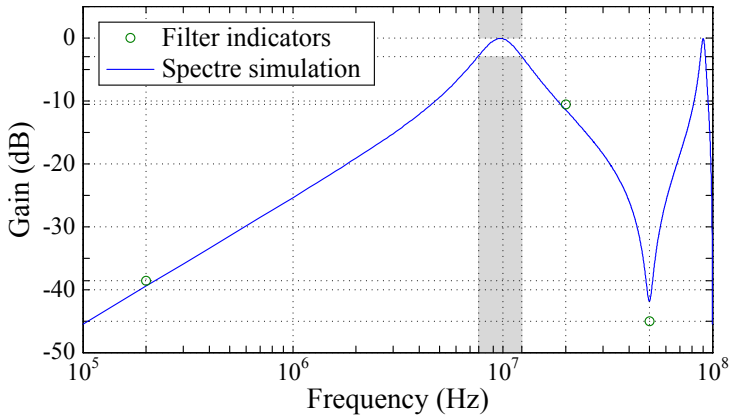


Figure 6.17: Frequency response of the best chromosome once the optimization process is complete.

6.2.4 Example III: Sixth-Order Bilinear Bandpass SC Filter

Although the hierarchical methodology works well for biquad sections, when optimizing two or more cascaded sections, the optimization time, even with varying goal specifications, greatly increases, due to the large number of variables in the design space.

To overcome this issue, when designing high-order filters, the software generates individual specifications for each biquad section based on the high-order filter specifications, the filter's order and the type of filter (e.g. bandpass, lowpass). Each section is then individually optimized based on

[3] (A) Specifications; (B) Equations/Spectre using ideal components; (C) Spectre - Real amplifier and ideal switches; (D) Ngspice - Real circuit before parasitic compensation; (E) Ngspice - Real circuit after (fine-tuning) optimization complete; (F) Spectre - Real circuit after (fine-tuning) optimization complete.

the generated biquad specifications, i.e., the optimization procedure shown in Fig. 6.9 is repeated for each biquad section, except for the top-level adjustment (transient simulations) step, which is only performed for the complete high-order filter once all biquad sections are individually optimized.

By optimizing each biquad section individually, the complexity of the problem is reduced (smaller, more manageable design spaces), allowing each section's design space to be fully explored, greatly decreasing the time it takes to find a solution within the desired high-order specifications.

In this section, an example of a sixth-order bandpass SC filter is presented to demonstrate how the biquad specifications are generated and to shown the performance of optimization software when optimizing high-order filters. The biquad section used to implement the sixth-order bandpass SC filter is shown in Fig. 6.18. The filter architecture is based on the continuous-time bandpass Sallen-Key topology [35], replacing the resistors with parallel SC branches, except for the feedback resistor, which is implemented using a bilinear network. Capacitor C_7 and C_9 are used to allow the compensation of the amplifier's parasitic capacitances. Capacitor C_6 introduces an additional zero to the filter's transfer function, allowing the implementation of bilinear transform based discrete filters. Capacitors C_4, C_5, C_{11}, and C_{12} are used to reduce the ratio between the largest and the smallest capacitor [6].

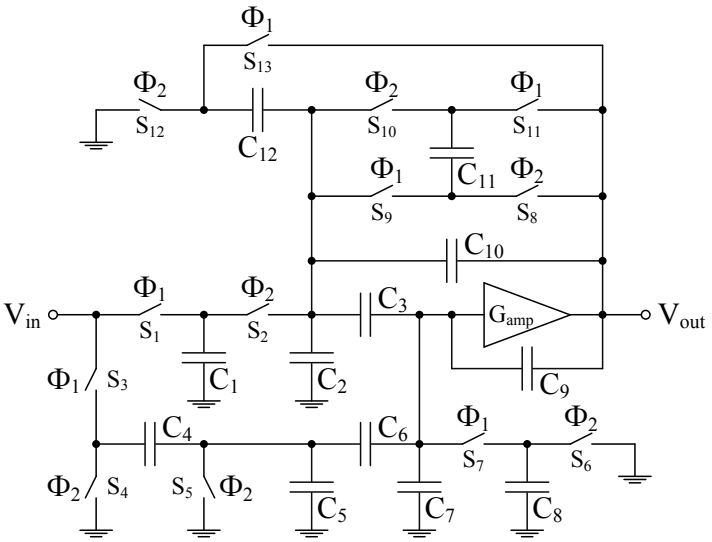


Figure 6.18: Bilinear bandpass SC biquad section used to implemented the sixth-order filter.

Note that the capacitor in the bilinear network (C_{11}) changes polarity depending on the active clock phase. Since the filter is implemented without the use of a CMFB circuit, there is going to be a significant voltage difference between each terminal of the capacitor. Every time the clock phase changes, there is going to be a large current flowing through the capacitor to compensate the voltage difference introduced by the rotation of the capacitor, which will lead to stability issues. To avoid this problem, since the filter is implemented in differential configuration, instead of rotating the capacitor between phases, capacitors C_{11} will change between the top and bottom part of the differential circuit every time the phase changes, producing the same effect while avoiding the significant voltage differences when the phase changes.

Using equation (6.13), it is possible to obtain individual specifications for the top-level optimization of each biquad section, based on the sixth-order specifications.

$$\left.\begin{cases} f_{c_{1,2}} = f_c \mp (B/4 + B/8 + B/16) & f_{c_3} = f_c \\ f_{pl_{1,2}} = f_{c_{1,2}} - (B/4 + B/8 + B/16)/2 & f_{pl_3} = f_{c_3} - B/2 \\ f_{ph_{1,2}} = f_{c_{1,2}} + (B/4 + B/8 + B/16)/2 & f_{ph_3} = f_{c_3} + B/2 \\ A_{max1_{1,2}} = A_{max1} - 0.25 & A_{max1_3} = 2 \times A_{max1_1} + \frac{\triangle A_{max2,1_1} \triangle f_{c_{3,1}}}{f_{ph_1} - f_{c_1}} + \frac{\triangle A_{max1,2_1} \triangle f_{c_{2,3}}}{f_{c_2} - f_{ph_2}} \\ A_{max2_{1,2}} = A_{max2} + 0.50 & A_{max2_3} = A_{max1_3} - |A_{max1_1} - A_{max2_1}| \end{cases}\right\} \begin{matrix}\text{Biquad 1 and 2} \\ \text{Biquad 3}\end{matrix} \quad (6.13)$$

where, A_{max_1} and A_{max_2} represent the attenuation limits within the passband, f_{pl} and f_{ph} the lower and higher passband frequencies, respectively, B the passband bandwidth ($f_{ph} - f_{pl}$), and f_c the central frequency. The constants in $A_{max_{1_{1,2}}}$ and $A_{max_{2_{1,2}}}$ are used to increase the chances of the sixth-order filter fitting inside the passband specifications when the biquad sections are cascaded together.

During the top-level optimization, the chromosomes of each section were optimized against capacitor mismatch variations, during 50 generations of 5000 chromosomes, using the biquads' differential transfer function. The first section proved to be the most sensitive to random variations, having a 90 % probability of failing the passband specification before the mismatch optimization and 83 % afterwards, the second section improved from 31 % to 17 %, and the third section from 80 % to 36 %.

Since the filter's input and output common-mode voltages are different, the biquad sections use complementary amplifiers, i.e., the odd numbered sections use a voltage-combiner with a NMOS differential pair, pushing the common-mode up ($V_{cmo} > V_{cmi}$), and the even numbered section uses a voltage-combiner with a PMOS differential pair, pushing the common-mode down ($V_{cmo} < V_{cmi}$). Using the variation between input and output common-mode voltages in the amplifier's cost function, the amplifiers can be optimized to introduce approximately the same absolute variation, maintaining the average common-mode voltage between each two sections at $V_{DD}/2$.

The amplifier's closed-loop transfer function, in each section, was calculated in the same manner as in the previous examples, considering the signal-flow diagram shown in Fig. 6.19 for clock phase Φ_1.

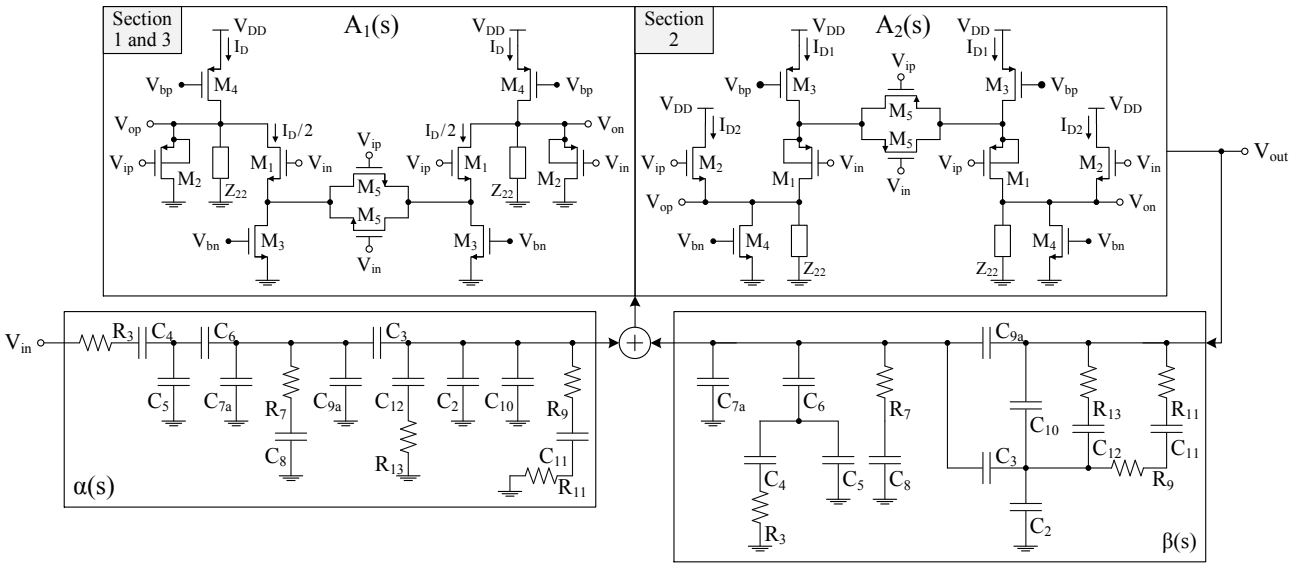


Figure 6.19: Closed-loop diagram of the bandpass SC filter during phase Φ_1.

where,

$$Z_{22}^{\Phi_1} = \left(\left[\frac{1}{s\,C_2} \parallel \frac{1}{s\,C_3} \right] + \left[\left(\frac{1}{s\,C_{10}} \right) \parallel \left(\frac{1}{s\,C_{11}} + R_9 + R_{11} \right) \parallel \left(\frac{1}{s\,C_{12}} + R_{13} \right) \right] \right) \parallel \left(\frac{1}{s\,C_9} \right) \tag{6.14}$$

and,

$$\begin{aligned} A_1(s): \quad & C_{7a} = C_7 - C_{gb_1} - (1+G_1)C_{gd_1} - (1+G_2)C_{gs_1} - C_{gd_2} - C_{gg_5} \\ & C_{9a} = C_9 - C_{gb_2} - C_{gs_2} \\ A_2(s): \quad & C_{7a} = C_7 - (1+G_2)(C_{gb_1} + C_{gs_1})(1+G_1)C_{gd_1} - C_{gb_2} - C_{gd_2} - C_{gg_5} \\ & C_{9a} = C_9 - C_{gs_2} \end{aligned} \tag{6.15}$$

Table 6.19 shows the results obtained from the amplifier optimization step, before and after the PVT optimization (Type III). This optimization starts once the amplifier specifications are met, using a smaller population (500 chromosomes). During the first 50 generations, the PVT fitness is calculated using equation (6.3) with a weight of 1, i.e., considering only the performance of the worst corner. Afterwards, the weight is reduced to 95 % and the optimization stops once the solution of the typical corner has the desired gain value. This is done to find a solution with the correct gain and as robust as possible against PVT corners. The optimization results show that the software improved the amplifier's gain and settling time robustness to random variations at the cost of a lower linearity within the input voltage range.

Table 6.19: Amplifier's parameters specifications, mean value, and standard deviation before and after the PVT optimization: (S1) section 1, (S2) section 2, (S3) section 3.[4]

		Gain			Linearity ($V_{amp_{in}} = 100$ mV)			Settling Time		
		G (V/V)	μ (V/V)	σ (mV/V)	Lin. (mV/V)	μ (mV/V)	σ (mV/V)	ST (%)	μ (%)	σ (%)
S1	(A)	1.450	-	Min.	≤ 1.00	-	Min.	≤ 60.00	-	Min.
	(B)	1.450	1.440	72.47	0.54	1.94	1.63	58.92	58.20	8.96
	(C)	1.451	1.450	17.21	9.73	11.25	3.85	74.75	74.35	2.63
S2	(A)	1.370	-	Min.	≤ 1.00	-	Min.	≤ 60.00	-	Min.
	(B)	1.370	1.320	108.99	1.08	3.03	2.67	43.89	40.37	9.09
	(C)	1.371	1.316	77.83	2.52	9.34	6.75	46.29	41.34	6.89
S3	(A)	1.350	-	Min.	≤ 1.00	-	Min.	≤ 60.00	-	Min.
	(B)	1.350	1.344	59.50	0.88	2.41	2.20	53.11	53.49	11.20
	(C)	1.351	1.333	52.66	8.50	10.74	5.07	53.51	50.32	9.05

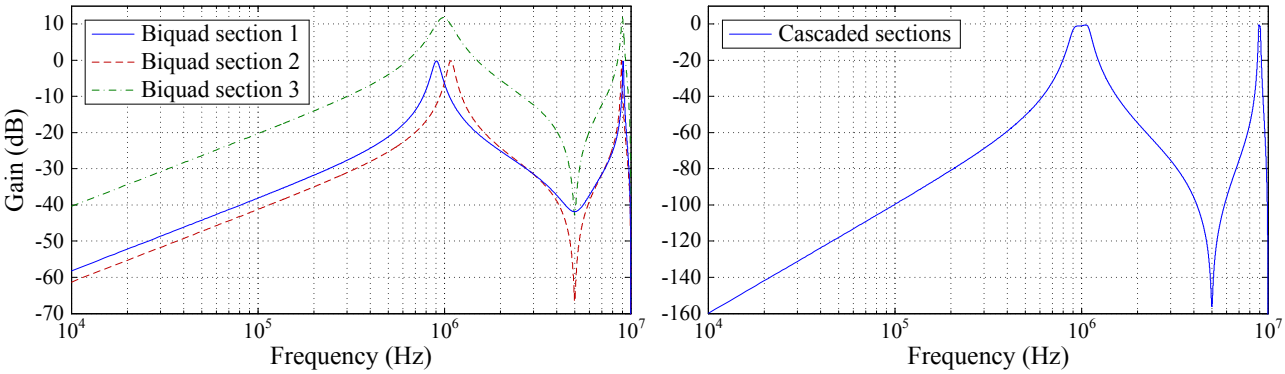
The optimization results of each section are shown Table 6.20. Using a smaller passband in terms of gain in the first two sections, it was possible for the cascaded sections to fit within the desired specifications in the first generation of the top-level adjustment step, leaving the remaining generations to fine-tune the capacitor values due to the effect of the switches parasitic capacitances.

[4] (A) Specifications; (B) Ngspice - Real circuit before PVT optimization; (C) Ngspice - Real circuit after PVT optimization.

Table 6.20: Specifications and simulation results of the bandpass SC filter after the optimization process is complete: (S1) section 1, (S2) section 2, (S3) section 3, (CS) cascaded sections.[5]

	$f_{stoplow}$ (kHz)	$f_{passlow}$ (kHz)	$f_{central}$ (kHz)	$f_{passhigh}$ (kHz)	$f_{stophigh}$ (kHz)	$F_s/2$ (kHz)
S1	9.125	868.750	912.500	956.250	1912.500	5000.000
(A)	Minimize	$\geq$ -3.00 dB	$\leq$ 0.00 dB	$\geq$ -3.00 dB	Minimize	Minimize
(B)	-59.00 dB	-2.22 dB	-0.26 dB	-2.98 dB	-24.34 dB	-41.85 dB
S2	10.875	1043.750	1087.500	1131.250	2078.500	5000.000
(A)	Minimize	$\geq$ -3.00 dB	$\leq$ 0.00 dB	$\geq$ -3.00 dB	Minimize	Minimize
(B)	-60.57 dB	-2.42 dB	-0.14 dB	-2.83 dB	-24.85 dB	-66.72 dB
S3	10.000	900.000	1000.000	1100.000	2000.000	5000.000
(A)	Minimize	$\geq$ 9.00 dB	$\leq$ 12.00 dB	$\geq$ 9.00 dB	Minimize	Minimize
(B)	-40.38 dB	9.74 dB	11.74 dB	9.24 dB	-5.45 dB	-42.79 dB
CS	100.000	900.000	1000.000	1100.000	2000.000	5000.000
(A)	Minimize	$\geq$ -3.50 dB	$\leq$ 0.25 dB	$\geq$ -3.50 dB	Minimize	Minimize
(B)	-99.70 dB	-3.40 dB	-1.22 dB	-3.12 dB	-54.73 dB	-89.33 dB

The frequency responses obtained from a Spectre simulation of the impulse response for each biquad section and of the three cascaded sections are shown in Fig. 6.6.



(a) (b)

Figure 6.20: Frequency response obtained from Spectre: (a) biquad sections, (b) cascaded sections.

The design obtained from the optimization software after the top-level adjustment step is shown in Table 6.21. Note that, in this example, the chromosome has two genes to individually control the diode connected transistors' current used to bias M_3 and M_4, while in the previous examples the current in the diode connected transistor biasing M_3 had a fixed 1/2 ratio in relation to the one biasing M_4.

On average, using a single core of the Intel Xeon X5570 processor (2.93 GHz) in a shared, i.e., multi-user, server, the top-level optimization step takes 20 minutes/section and an additional 20 minutes/section if capacitor mismatch optimization is performed. The amplifier takes 15 minutes/section to be optimized and an additional 30 minutes/section if PVT optimization is performed. In total, using the approach described in this section, the software takes between 4 to 5 hours to obtain an optimized solution, within the desired specifications.

[5] (A) Specifications; (B) Spectre - Real circuit after (fine-tuning) optimization complete.

Table 6.21: Design obtained for each biquad section after the optimization procedure ends: (S1) section 1, (S2) section 2, (S3) section 3.

	C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)
S1	30.1	620.8	2146.0	121.0	297.4	407.6
S2	25.0	485.6	1935.3	226.3	223.5	150.2
S3	112.4	273.8	972.4	337.0	26.0	138.9
	C_{7c} (fF)	C_{8c} (fF)	C_{9c} (fF)	C_{10c} (fF)	C_{11c} (fF)	C_{12c} (fF)
S1	140.4	76.8	635.0	100.0	2327.5	1291.9
S2	54.5	133.1	1222.4	137.8	1160.4	109.0
S3	51.0	40.3	545.0	104.2	633.3	233.2
	W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	I_{D1} (µA)
S1	48.13	133.31	122.08	113.52	25.39	43.53
S2	49.98	56.36	140.63	113.30	83.88	64.46
S3	15.07	145.94	149.52	5.28	3.43	101.94
	L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	I_{D2} (µA)
S1	0.58	0.76	0.29	0.65	0.89	76.59
S2	0.16	0.98	0.59	0.69	0.46	76.91
S3	0.46	0.59	0.64	0.20	0.42	86.27

V_{DD} (V)	V_{cmi1} (mV)	V_{cmi2} (mV)	V_{cmi3} (mV)	V_{cmo} (mV)	Power (µW)
1.20	450.0	786.1	453.8	778.8	1156.16

6.3 Non-Hierarchical Optimization Methodology

6.3.1 Method Description

The non-hierarchical optimization methodology, which uses the analysis method presented in Section 5.2, optimizes the frequency response of SC filters using a single system of (first-order differential) equations that completely describes the circuit's continuous-time behavior in any clock phase, independently of the number of phases that control the circuit.

To calculate the filter's frequency response from the system of differential equations, it is necessary to run DC OP simulations to extract the transistors' parameters (*gm*, *gds*, *gmb*, and parasitic capacitances) and to indicate the desired output node. The software can then perform all the necessary calculations to obtain the poles and the zeros of the discrete-time transfer function. Since the software is coded in C programming language, the matrix computations, which include basic operations with matrices and the calculation of eigenvalues, eigenvectors, and inverse matrix, are done using the numerical GNU scientific library (GSL) [85]. Using the numerical values of the poles and the zeros, it is possible to efficiently compute the numerical value of the SC filter's transfer function for any frequency value.

In this optimization methodology, besides the filter's frequency response, the software evaluates the total circuit area and amplifier's power consumption, output common-mode voltage, and linearity in the input voltage range.

Since the problem of optimizing a filter circuit is very difficult, the way the fitness is computed changes during the optimization procedure. The idea is to use progressively ”harder” levels for the computation of the chromosomes' fitness to minimize the time it takes to converge to a robust solution. The flow of the non-hierarchical methodology is shown in Fig. 6.21.

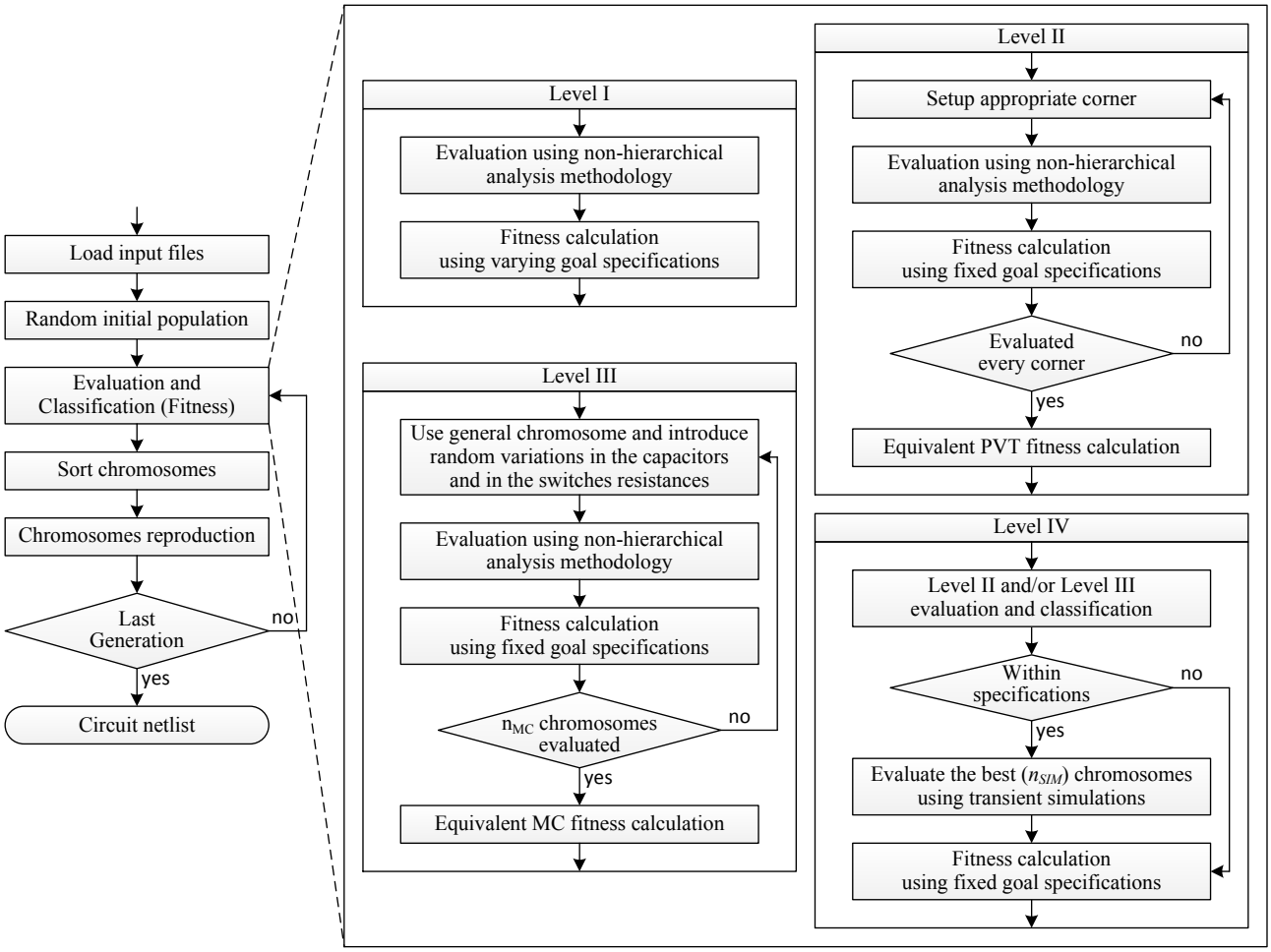


Figure 6.21: Flow of the non-hierarchical optimization methodology.

6.3.1.1 Level I: Standard optimization using varying goal specifications

Depending on the filter's size and specifications, it can be difficult (take more time) for the GA to converge to a good solution when compared with the hierarchical methodology, due to the size of the chromosome. Here, the chromosome is composed by all components in the circuit (capacitors, transistors, current and voltage sources), while in the hierarchical methodology the components were broken down into different chromosomes depending on the optimization step, resulting in smaller, more manageable design spaces.

To improve the convergence speed of the GA, it is especially important to use varying goal specifications (Type II) in the more demanding parameters, as described in Section 6.1.3.2. The software will remain in this optimization level during, at least, $gen_{varSpec}$ generations, using equation (6.16) to

calculate the chromosomes' fitness. After this number of generations, and if there at least one solution within the desired specifications, the software will change to the next optimization level, otherwise, the GA changes to equation (6.17), with fixed goal specifications, until finding a solution.

$$fitness = \underbrace{f_{max}(A_n \,\epsilon\, [A_{max1}, A_{max2}]) \times f_{equ}(A_{min1}) \times f_{equ}(A_{min2})}_{\text{frequency response indicators}} \times f_{min}(\Delta G) \times f_{min}(Area) \times f_{min}(Power) \times f_{equ}(V_{cmo}) \quad (6.16)$$

$$fitness = \overbrace{f_{max}(A_n \,\epsilon\, [A_{max1}, A_{max2}]) \times f_{max}(A_{min1}) \times f_{max}(A_{min2})} \times f_{min}(\Delta G) \times f_{min}(Area) \times f_{min}(Power) \times f_{equ}(V_{cmo}) \quad (6.17)$$

Note that, in the following optimization levels, since the GA is already close to the desired solution, the software will keep using equation (6.17), with fixed goal specifications (Type I).

6.3.1.2 Level II and III: Optimization under different PVT corners and with mismatch errors

Once the software enters these optimization levels, several chromosomes within the desired specifications should have already been found. However, they might be susceptible to PVT and MC variations (mismatch errors). To improve the solutions robustness to these variations, the chromosomes are optimized using the procedure described in Section 6.1.3.3.

The software can perform both of these optimizations individually, in any specific order, or concurrently. From user experience, due to the large number of parameters in the chromosome, trying to optimize chromosomes under different PVT corners and with mismatch variations at the same time leads to longer optimization times, since the software will have to adjust more components' values at once, making it harder to find the local minima in the design space. Since the filter's frequency response is more susceptible to the amplifier's (gain) PVT corner variations, when compared to the expected mismatch errors of the capacitors, it is recommended to start with the PVT optimization and, only after, introduce the mismatch errors, in the capacitors and in the switches ON resistance, for each corner of each chromosome, to minimize the time it takes to get a robust solution.

Once a solution is found within the desired specifications, even under different PVT corners and with mismatch errors, the software changes to the last optimization level.

6.3.1.3 Level IV: Hybrid optimization

When the software changes to this level, after the evaluation of each population under different PVT corners and with mismatch errors, using the analysis methodology from Section 5.2, the software evaluates, through transient simulations of the impulse response, the best (n_{SIM}) chromosomes of the current population, under the typical corner and without mismatch errors, and another n_{SIM} chromosomes created based on the best simulated-based genetic material from the previous generation.[6] Since these simulations are time consuming and the analysis method is very accurate, the value of n_{SIM} should be low (typically between 5 an 10).

[6] During the first generation in this optimization level, since there is no simulation-based genetic material from the previous generation, only the best n_{SIM} equation-based chromosomes are evaluated through simulation.

To distinguish chromosomes evaluated though simulation from the ones evaluated using equations, the fitness value of the simulation-based chromosomes, calculated from (6.17), is incremented by one, varying from one to two, instead of from zero to one.

This optimization level is used to validate the equation-based design and to fine-tune the capacitor values due to the effect of the switches' parasitic capacitances, that change over a clock period. Once a solution within the desired specifications is found, through simulation, the optimization procedure ends.

6.3.2 Example I: Second-Order Lowpass SC Filter

In this section, the second-order lowpass SC filter described in Section 5.2.4, reproduced in Fig. 6.22 for convenience, is used to demonstrate the optimization method described in Section 6.3.1 and to allow a performance comparison between the two optimization methods described in this chapter, using the same filter and under the same conditions. For simplicity purposes the circuit is shown in single-ended configuration and the parasitic capacitances are not shown.

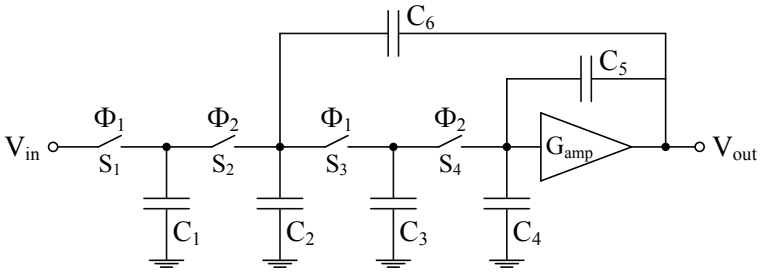


Figure 6.22: Second-order lowpass SC filter.

The circuit was designed in a standard 1.2 V 130 nm CMOS technology and with a clock frequency F_s = 100 MHz. The filter's frequency response was optimized using the design specifications shown in Table 6.22 and using the amplifier circuit shown in Fig. 6.11, while trying to minimize the total area and total power consumption.

Table 6.22: Specifications used to optimize the frequency response of the lowpass SC filter.

$F_{passlow}$ (F_{pl})	F_{cutoff} (F_c)	F_{stop}	$F_s/2$
10 kHz	1 MHz	6 MHz	50 MHz
0.01 dB $\geq$ $\left[F_{pl}, F_c\right]$ $\geq$ -3.01 dB		$\leq$ -30.00 dB	$\leq$ -59.00 dB

Several optimization runs were made in order to characterize the procedure, i.e., to determine reasonable values for the population size and number of generations used in each optimization level, without sacrificing the amount of time needed to find an optimized solution. The best chromosome of an arbitrary optimization run, which is shown in Table 6.23.

Table 6.23: Design obtained from a run without PVT corners or mismatch variations optimization.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)
91.0	353.5	50.0	497.4	126.1	6156.6
W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	I_D (µA)
0.30	90.98	30.90	33.00	1.89	91.0
L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	V_{DD} (V)
0.51	0.22	0.20	0.59	0.64	1.2
W_{S1} (µm)	W_{S2} (µm)	W_{S3} (µm)	W_{S4} (µm)	L_S (µm)	V_{cmi} (mV)
0.82	0.64	0.19	0.44	0.12	450.0

The comparison between the filter's frequency response obtained through a transient simulation of the impulse response in Spectre and from the optimization software using the non-hierarchical analysis methodology, without the optimization of PVT corners or mismatch variations, using the design from Table 6.23, is shown in Fig. 6.23.

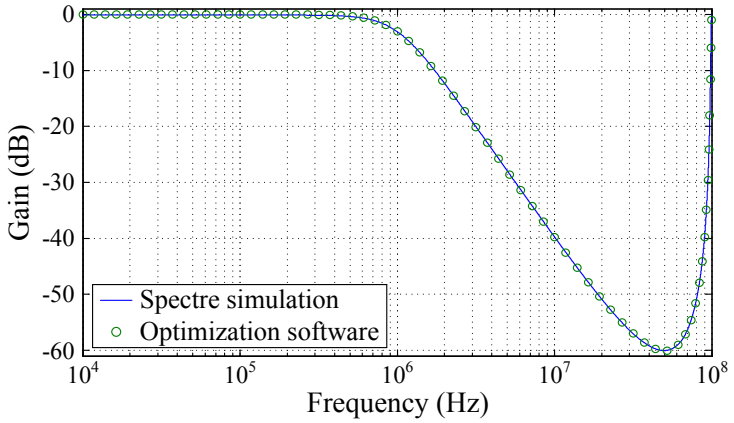


Figure 6.23: Frequency response of an arbitrary run once the optimization procedure is complete.

Table 6.24 shows the frequency response results of the lowpass SC filter obtained from the non-hierarchical optimization methodology and from transient simulations of the impulse response in Spectre and in Ngspice. The results show that the values obtained from the optimization software (row B) are in good agreement with the Spectre and Ngspice simulations of the complete filter circuit. From the Spectre (row C) and Ngspice (row D) simulations, it is also possible to conclude that both simulators produce similar results.

Table 6.24: Specifications and frequency response results of the lowpass SC filter obtained from different methods.[7]

	$F_{passlow}$	F_{cutoff}	F_{stop}	$F_s/2$
	10 kHz	1 MHz	6 MHz	50 MHz
(A)	$\leq$ 0.01 dB	$\geq$ -3.01 dB	$\geq$ -30.00 dB	$\leq$ -59.00 dB
(B)	-0.03 dB	-2.66 dB	-30.55 dB	-59.45 dB
(C)	-0.02 dB	-2.63 dB	-30.55 dB	-59.72 dB
(D)	-0.11 dB	-2.71 dB	-30.76 dB	-59.66 dB

[7] (A) Specifications; (B) Equations using non-hierarchical analysis methodology; (C) Spectre - Real circuit after (fine-tuning) optimization complete; (D) Ngspice - Real circuit after (fine-tuning) optimization complete.

A new run was done to evaluate the performance of the non-hierarchical optimization methodology with PVT corners and mismatch variations. The frequency response of the best chromosome (for 10 different cases of random variations), in a random corner (not the typical) and at different levels of the optimization procedure, is shown in Fig. 6.24 to Fig. 6.26. Results show that the PVT optimization prioritizes solutions further away from the forbidden areas to avoid failing any of the specifications in the worst corners and the mismatch optimization reduces the variability to the random variations.

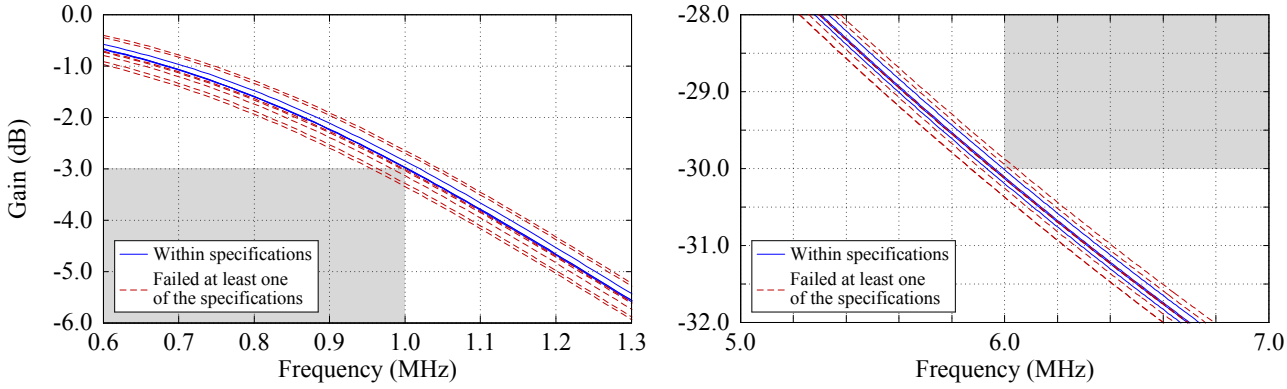


Figure 6.24: Frequency response of the best chromosome (a) around the passband frequency and (b) around the stopband frequency before the PVT corners and mismatch variations optimization.

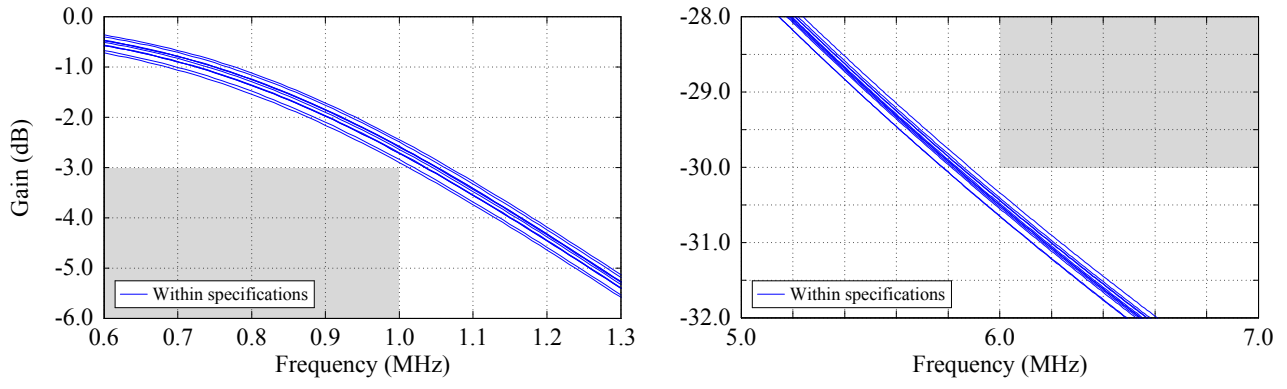


Figure 6.25: Frequency response of the best chromosome (a) around the passband frequency and (b) around the stopband frequency after the PVT corners optimization.

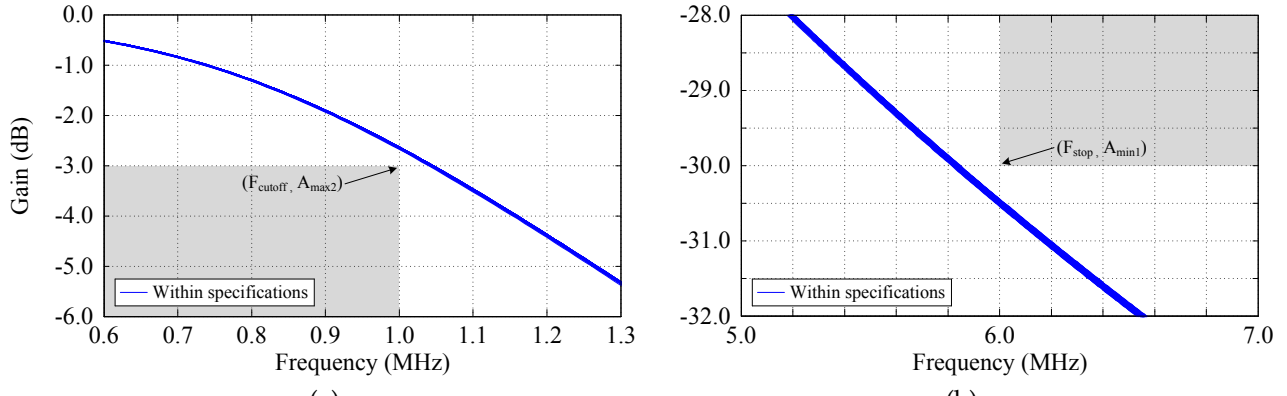


Figure 6.26: Frequency response of the best chromosome (a) around the passband frequency and (b) around the stopband frequency after the PVT corners and mismatch variations optimization.

The PVT optimization was performed with 150 individuals during 25 generations ($\sim$ 15 minutes) and the mismatch optimization with 500 individuals for the same number of generations ($\sim$ 15 minutes). In this example, two different types of random variations were used in the Gaussian variable 3σ, 10 % to scale the main capacitors in the filter, which allows the software to detect and discard solutions with incomplete settling, and 0.5 % to introduce random variations in each capacitor of the differential circuit. Table 6.25 shows the final design obtained from the PVT corners and mismatch variations optimization, which fits within the desired specifications, even with these variations.

Table 6.25: Design obtained from a run with PVT corners and mismatch variations optimization.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)
115.4	549.7	50.0	464.3	434.3	7179.7
W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	I_D (µA)
1.17	96.15	34.91	68.36	0.17	101.0
L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	V_{DD} (V)
0.93	0.17	0.25	0.21	0.18	1.2
W_{S1} (µm)	W_{S2} (µm)	W_{S3} (µm)	W_{S4} (µm)	L_S (µm)	V_{cmi} (mV)
0.62	0.58	0.29	0.90	0.12	450.0

6.3.2.1 Software performance test: Hierarchical Vs. Non-hierarchical methodology

In this section, a performance comparison is made between the hierarchical and the non-hierarchical optimization methodologies presented in this chapter, without PVT corners or mismatch variations optimization, with the circuit used in Section 6.2.2 and Section 6.3.2. This comparison is made under the same conditions, i.e., the same SC filter, design space, filter specifications, number of individuals (1000), number of generations using varying goal specifications (30), and number of points used to evaluate the filter's passband and the gain within the amplifier's input voltage range. Since the hierarchical methodology can have convergence problems due to the multi-step nature of the method, the maximum optimization time to find a solution within the desired specifications was set to one hour.

The average value of several parameters, obtained after performing 30 optimization runs of each method, are shown in Table 6.26. Results show that hierarchical methodology is faster, since the optimization problem is divided into smaller problems with less variables, i.e., more manageable design spaces, making it easier to find solutions, in the design space, within the desired specifications, while minimizing other parameters, such as the area occupied by the components.

However, the non-hierarchical methodology allows amplifiers to be designed with minimum power consumption, since this method accurately models the non-ideal effects of the circuit's components, so the amplifiers can be safely designed without concerns of having incomplete settling in any of the nodes in the circuit. Since the hierarchical method is based on approximated equations, the circuits will have to be designed by excess, leading to a larger than necessary power consumption in the amplifier.

In the hierarchical method, since each step is dependent on the previous ones, there is the risk of one step resulting in a specification that is not possible to achieve by the next step and the optimization procedure gets stuck, e.g. the values chosen for the capacitors, which are seen as a load by the amplifier, can be too large so that, for the amplifier's design space, it is impossible to obtain a closed-loop bandwidth high enough to charge the capacitors in the phases' duration, i.e., there would be incomplete settling, introducing unpredictable changes in the filter's transfer function with the equations used. In the non-hierarchical method, since the filter's transfer function is calculated in a single step, these solutions can be discarded based on the filter's frequency response.

Table 6.26: Comparison between non-hierarchical and hierarchical methods after 30 optimization runs.

	Non-hierarchical		Hierarchical [5]	
	μ	σ	μ	σ
Number of generations to find final solution	33	16	44	24
Amplifier current (μA)	201.8	45.9	255.6	32.4
Area occupied by the amplifier (μm^2)	125.5	36.1	134.8	54.8
Area occupied by the switches (μm^2)	0.487	0.120	0.245	0.106
Filter capacitance (pF)	13.30	2.15	11.88	1.69
Optimization time to find final solution (s)	489	-	268	-
Number of unoptimized solutions after one hour	0	-	3	-
Optimization time with failures (s)	489	-	601	-

6.3.3 Example II: Second-Order Bandpass SC Filter

In this section, the second-order bandpass SC filter from Section 5.2.3, shown in Fig. 6.27, is used as another demonstrator for the optimization method described in Section 6.3.1. For simplicity purposes the circuit is shown in single-ended configuration and the parasitic capacitances are not shown.

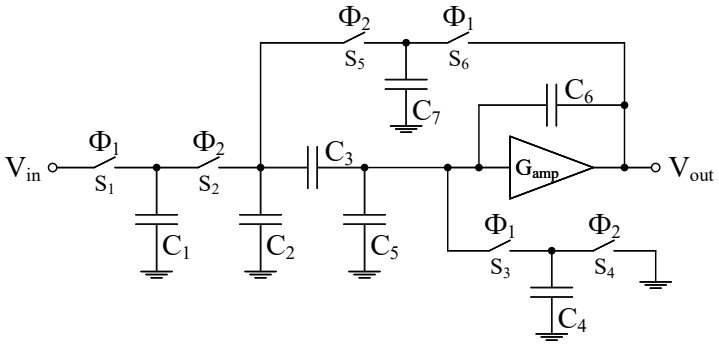


Figure 6.27: Second-order bandpass SC filter.

The SC filter was designed for a standard 1.2 V 130 nm CMOS technology and a clock frequency F_s = 10 MHz, for the design specifications shown in Table 6.27. To make the optimization more complex and see if the GA can still converge to a solution, the output common-mode voltage of the circuit is also used in the cost function. In this example, the software should minimize the difference between input and out common-mode voltages. For this reason, a common-drain amplifier is used at the voltage-combiner's output, as shown in Fig. 6.28, to move the common-mode back to $V_{DD}/2$.

Table 6.27: Specifications used to optimize the frequency response of the bandpass SC filter

$F_{stoplow}$	$F_{passlow}$ (F_{pl})	$F_{passhigh}$ (F_{ph})	$F_{stophigh}$	$F_s/2$
1.25 kHz	75.0 kHz	125.0 kHz	500.0 kHz	5.0 MHz
$\leq$ -40.50 dB	0.05 dB $\geq$ $[F_{pl}, F_{ph}]$ $\geq$ -3.05 dB		$\leq$ -17.00 dB	$\leq$ -32.50 dB

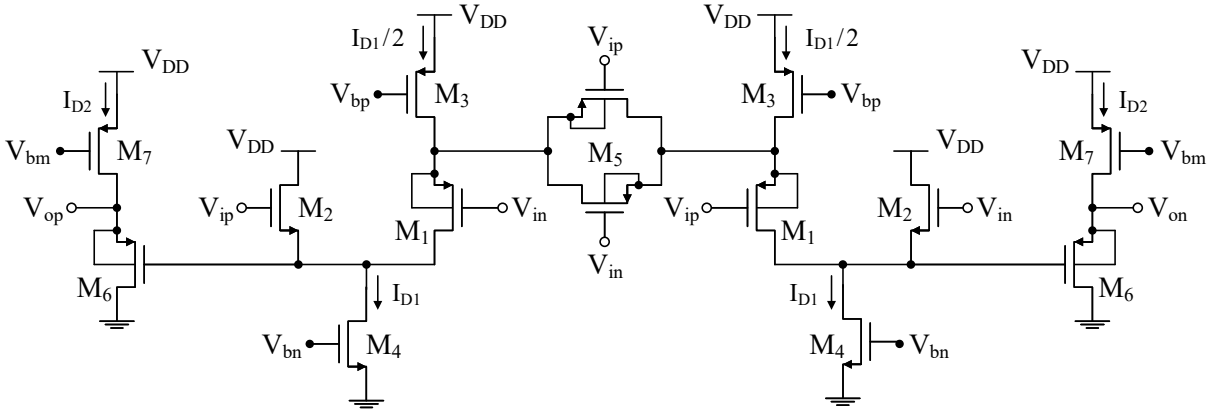


Figure 6.28: Differential voltage-combiner amplifier with source degeneration with a DC level shifter.

The bandpass SC filter was optimized using varying goal specifications during the first 30 generations, with a population size of 1000 individuals. Several optimizations runs were made to determine the amount of time the GA takes to converge to a solution. The average value of the optimization time and of some other parameters used in the cost function are shown in Table 6.28.

Table 6.28: Average results and standard deviation after performing 30 optimization runs

	Generations	Power (µW)	Amplifier Area (µm^2)	Total Filter Cap. (pF)	Opt. Time (min)
μ	58	417.5	345.5	14.0	29
σ	29	27.3	74.9	1.5	14

The best chromosome of an optimization run is shown in Table 6.29, and the resulting frequency responses of the bandpass SC filter, obtained from the optimization software using equations and through transient simulations in Spectre and in Ngspice are shown in Table 6.30. Results show that the optimization software gives approximate values to the ones obtained from transient simulations, with a maximum difference of 0.2 dB from the Spectre simulation.

Table 6.29: Design obtained from a run without PVT corners or mismatch variations optimization.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)	C_{7c} (fF)
40.1	953.7	4547.4	99.8	25.0	1123.3	239.0
W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	W_6 (µm)	W_7 (µm)
63.70	10.00	44.90	7.50	75.80	96.00	4.90
L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	L_6 (µm)	L_7 (µm)
0.84	0.24	0.60	0.84	0.72	0.84	0.60
W_s (µm)	L_s (µm)	I_{D1} (µA)	I_{D2} (µA)	V_{DD} (V)	V_{cmi} (mV)	V_{vcmo} (mV)
1.40	0.12	78.4	54.8	1.20	600.0	594.7

Table 6.30: Specifications and frequency response results of the bandpass SC filter obtained from different methods.[8]

	$F_{stoplow}$	$F_{passlow}$	$F_{central}$	$F_{passhigh}$	$F_{stophigh}$	$F_s/2$
	1.25 kHz	75.0 kHz	100.0 kHz	125.0 kHz	500.0 kHz	5.0 MHz
(A)	$\leq$ -40.50	$\geq$ -3.05	$\leq$ 0.05	$\geq$ -3.05	$\leq$ -17.00	$\leq$ -32.50
(B)	-42.81 dB	-2.93 dB	-0.61 dB	-3.05 dB	-19.06 dB	-35.43 dB
(C)	-42.78 dB	-3.01 dB	-0.51 dB	-2.85 dB	-18.99 dB	-35.31 dB
(D)	-42.87 dB	-2.96 dB	-0.63 dB	-3.03 dB	-19.17 dB	-35.53 dB

Afterwards, a new optimization run was performed with PVT corners and mismatch variations turned on. The PVT optimization (of 21 corners) was performed with 150 individuals during 50 generations ($\approx$ 75 minutes) and the mismatch optimization with 500 individuals during the same number of generations ($\approx$ 35 minutes). The frequency responses of the best chromosome, for 10 cases of random variations, before and after the PVT and mismatch optimization are shown in Fig. 6.29 to Fig. 6.31. Once again, the results show that the variability of the best chromosome reduces with these optimizations.

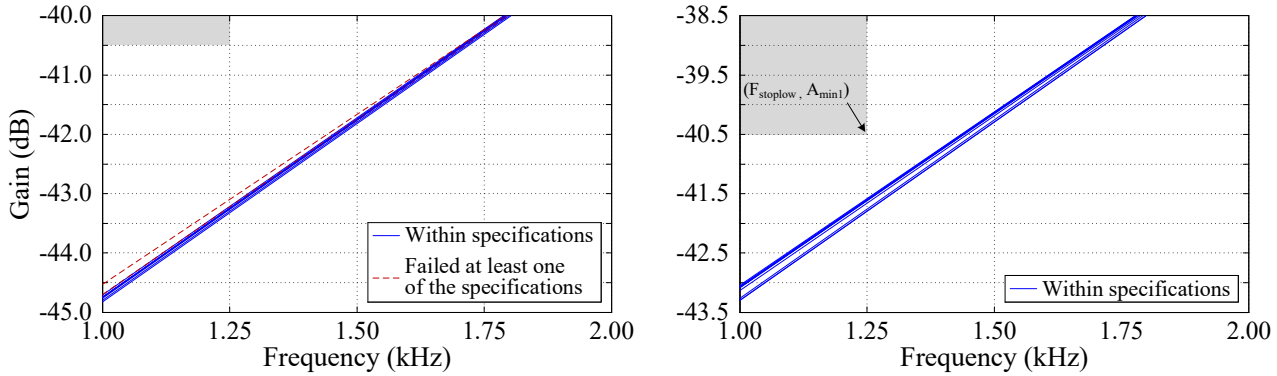


Figure 6.29: Frequency response of the best chromosome around the lower stopband frequency (a) before and (a) after the PVT corners and mismatch variations optimization.

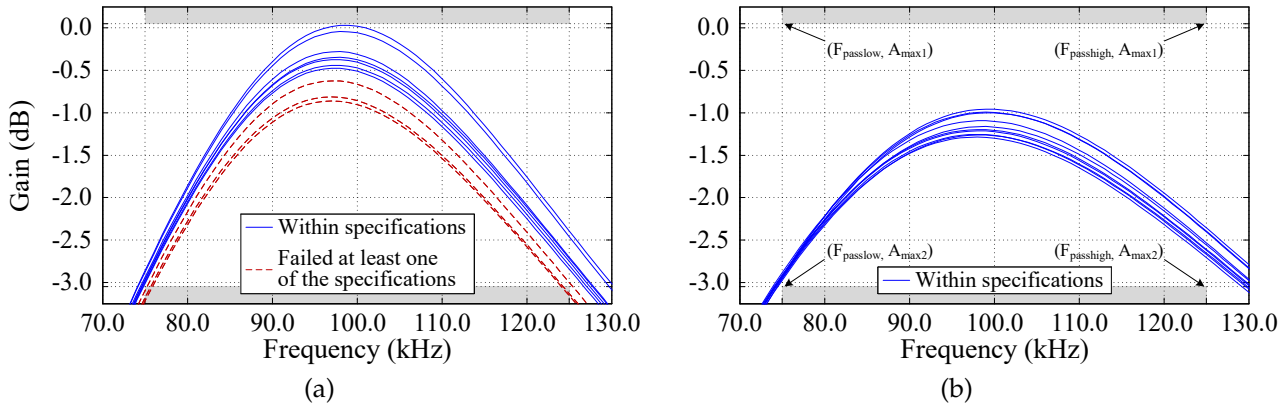


Figure 6.30: Frequency response of the best chromosome around the central frequency (a) before and (a) after the PVT corners and mismatch variations optimization.

[8] (A) Specifications; (B) Equations using non-hierarchical analysis methodology; (C) Spectre - Real circuit after (fine-tuning) optimization complete; (D) Ngspice - Real circuit after (fine-tuning) optimization complete.

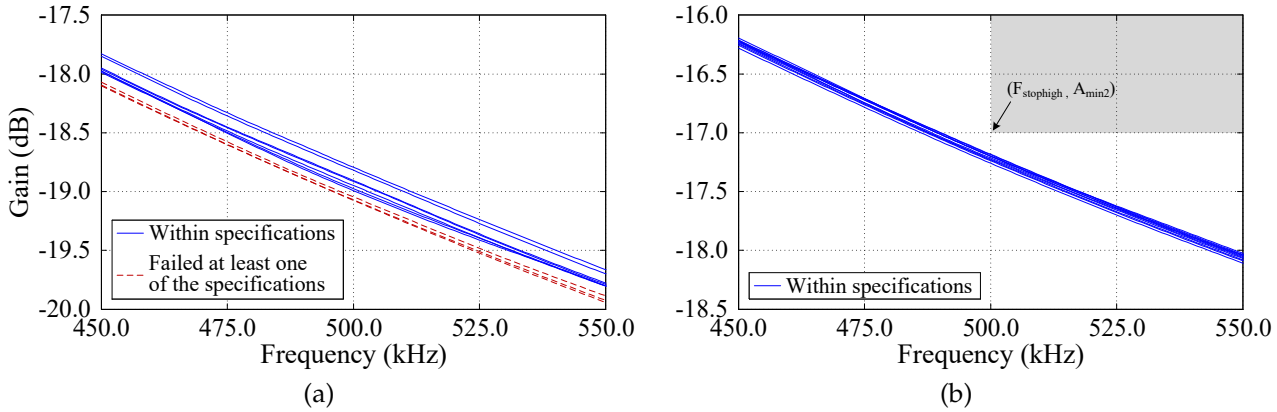


Figure 6.31: Frequency response of the best chromosome around the higher stopband frequency (a) before and (a) after the PVT corners and mismatch variations optimization.

The best chromosome of an optimization run, obtained after the PVT corners and mismatch variations optimization, is shown in Table 6.31, and the resulting frequency responses of the bandpass SC filter, obtained from the optimization software using equations and through transient simulations in Spectre are shown in Fig. 6.32.

Table 6.31: Design obtained from a run with PVT corners and mismatch variations optimization.

C_{1c} (fF)	C_{2c} (fF)	C_{3c} (fF)	C_{4c} (fF)	C_{5c} (fF)	C_{6c} (fF)	C_{7c} (fF)
45.3	1008.6	3737.0	70.9	25.0	1643.7	175.2
W_1 (µm)	W_2 (µm)	W_3 (µm)	W_4 (µm)	W_5 (µm)	W_6 (µm)	W_7 (µm)
99.50	13.10	23.10	87.40	63.60	77.50	66.90
L_1 (µm)	L_2 (µm)	L_3 (µm)	L_4 (µm)	L_5 (µm)	L_6 (µm)	L_7 (µm)
0.72	0.48	0.48	0.84	0.96	0.72	0.60
W_s (µm)	L_s (µm)	I_{D1} (µA)	I_{D2} (µA)	V_{DD} (V)	V_{cmi} (mV)	V_{vcmo} (mV)
1.00	0.12	92.6	56.4	1.20	600.0	604.5

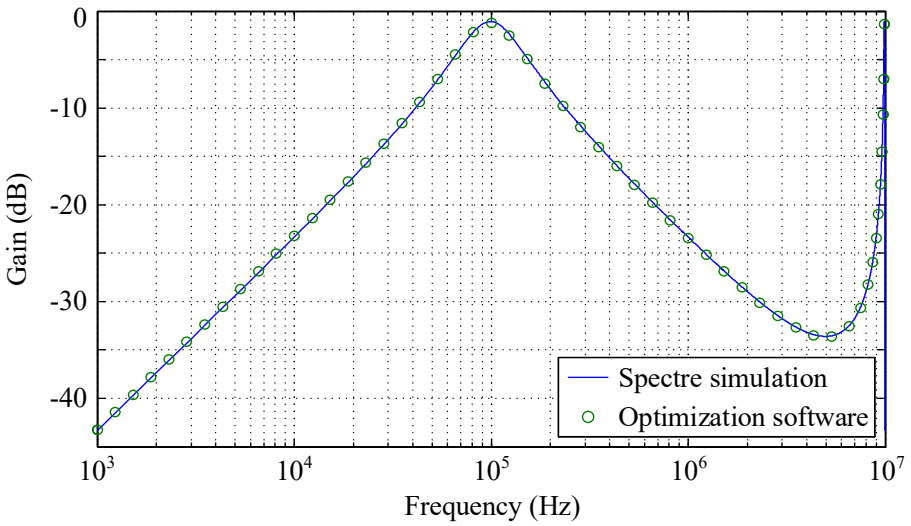


Figure 6.32: Frequency response of an optimization run once the optimization procedure is complete.

6.4 Conclusions

In this chapter, the optimization procedure using GAs and two equation-based optimization methodologies for the design of SC filters were described. In the first method the circuit is optimized using a multi-step approach, resulting in the overall design space being broken down into smaller, more manageable design spaces, making it easier to find solutions within the desired specifications. However, since this approach relies on approximated equivalent circuits/equations, non-ideal effects of some circuit components may not be correctly modeled, leading to amplifier designs with a higher power consumption than necessary and larger switches to safely avoid situations of incomplete settling in any of the nodes in the circuit. Due to the multi-step nature of this method, when chromosomes are optimized to increase their robustness under different PVT corners, the software can find solutions that are more robust to these variations, but that do not have exactly the desired amplifier gain. Since the filter is very sensitive to gain variations, a small variation in the gain can cause significant changes in the filter's frequency response, which would have to be compensated in the last, more time consuming, simulation-based optimization step or discarded for a less robust solutions but with the correct gain value.

In the second method, the circuit is optimized using a single system of first-order differential equations, that accurately model the non-ideal behavior of every component in the SC filter, allowing the amplifier circuit to be safely designed with minimum power consumption without concerns of having incomplete settling in the circuit. Due to the size of the design space, which includes every circuit component right from the beginning of the optimization procedure, it is imperative to use varying goal specifications in the filter's frequency response indicators, to increase the convergence speed of the GA. To keep the optimization time to a minimum, the PVT corners optimization should be perform first and only afterwards include mismatch variations for each corner, or vice-versa. This is done since, due to the large size of the design space, when a chromosome is not robust to the variations, if both of these optimizations are performed at the same time, the GA will have difficulty adjusting several components at the same time.

CHAPTER 7

SWITCHED-CAPACITOR FILTER PROTOTYPES AND MEASUREMENT RESULTS

In this chapter, the prototypes of a programmable bandpass switched-capacitor (SC) biquad filter using a low gain amplifier (Section 7.1) and of a 50 Hz notch SC filter using high gain amplifiers (Section 7.2) are presented. These circuits were implemented with 0.9 V supply voltage in UMC 130 nm CMOS technology and designed based on the analysis and optimization methodologies described in Chapters 5 and 6. The circuits were realized to be a part of an analog-to-digital (A/D) acquisition channel used in an internet-of-things (IoT) water management sensor node [86]. The simplified block diagram of this acquisition channel is shown in Fig. 7.1.

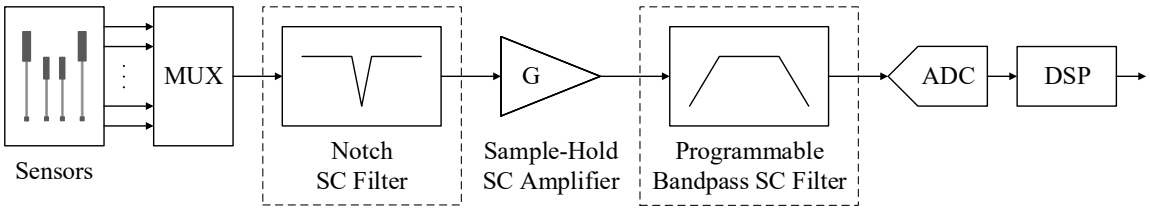


Figure 7.1: Simplified diagram of the acquisition channel used in a water management sensor node.

The information from the sensors (temperature, pressure, conductance, etc.) is a voltage that needs to be digitized, in order to be processed and uploaded into a high-level processing center. Most of these signals vary very slowly and can be considered as DC signals. The exception is the measurement of the water conductance (conductivity sensor), which requires digitizing signals with frequencies in the kHz range. When measuring sensors that produce DC voltages, the bandpass SC filter needs to be bypassed, to avoid the attenuation of the voltage signal. For this reason, an enable bit is used to determine when the filter should be part of the channel. When the filter is bypassed, the circuit is turned off to reduce the channels overall power dissipation. The notch SC filter is also implemented with an enable bit to allow the circuit to be bypassed and turned off when its use is not required.

An example of the frequency response of a conductivity sensor of glass substrate, obtained via thermal atomic layer deposition of aluminium oxide is shown in Fig. 7.2 [87].

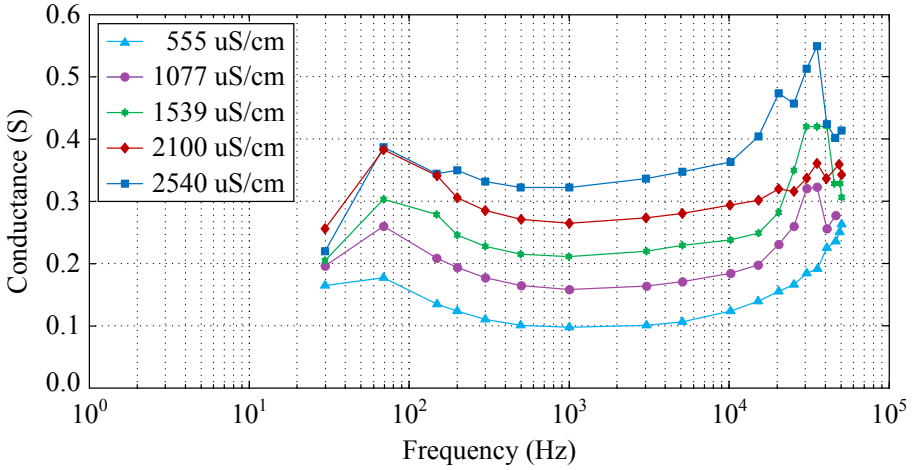


Figure 7.2: Frequency response of the sensor's conductance for different conductivity values.

These results show a linear variation in the sensors conductance with the water's conductivity for a frequency range between 200 Hz up to 10 kHz. Since the current used to drive the sensor is not fixed to a specific frequency value, the bandpass SC filter was made programmable with a central frequency between 4 kHz and 7 kHz, allowing some tunability in the sensor's bias current.

When deciding the type of analog-to-digital converter (ADC) and its required performance, it is important to take into consideration the overall performance of the channel. Considering the presence of unwanted signals, this means that the performance of the anti-aliasing filter (AAF) can be critical for the system. To facilitate the design and reduce the power dissipation of the AAF, it is possible to increase the sampling frequency, resulting in a reduction of the attenuation's requirements of the AAF. Since a sigma-delta modulator requires high oversampling ratio (OSR), which corresponds to a high sampling frequency, it is a better choice for the ADC topology than a successive approximation register (SAR), because the overall power dissipation of the channel (AAF + ADC) is lower. Additionally, a sigma-delta modulator can achieve a SNDR higher than 70 dB without requiring calibration schemes. For this reason, the sampling frequency of the modulator was selected as F_s = 2 MHz, allowing the AAF to be implemented with a simple RC circuit, while still obtaining more than 40 dB of alias suppression, and as F_s = 1 MHz for the remaining circuits (notch filter, S/H amplifier, bandpass filter).

The water network frequently shares its location with power network cables. Considering that the different sensors used for measuring the water quality can be connected through long cables to the digitizing channel, there will be coupling from the power supply signal into the sensors' cable. This means that a 50 Hz (EU) sine wave will be added to the desired signal from the sensor and, when the cable from the sensor is close to the power lines, the amplitude of the unwanted signal can be much larger than the desired signal's amplitude. For this reason, a notch filter is used to remove this unwanted signal.

7.1 Programmable Second-Order Bandpass Switched-Capacitor Filter

The programmable bandpass SC biquad filter was implemented using the topology described in Section 5.2.3 and optimized using the non-hierarchical methodology from Section 6.3, where capacitors C_1, C_2, and C_5 were made programmable using the structure shown in Fig. 7.3(c) and the switches were implemented using transmission gates to minimize the non-linearity of the switches' resistance. The differential capacitors are implemented using two capacitors in parallel, as shown in Fig. 7.3(b). This was done so that the same amount of parasitic capacitances are added to each side of the differential circuit, since the capacitor's top and bottom plate parasitic capacitances are significantly different.

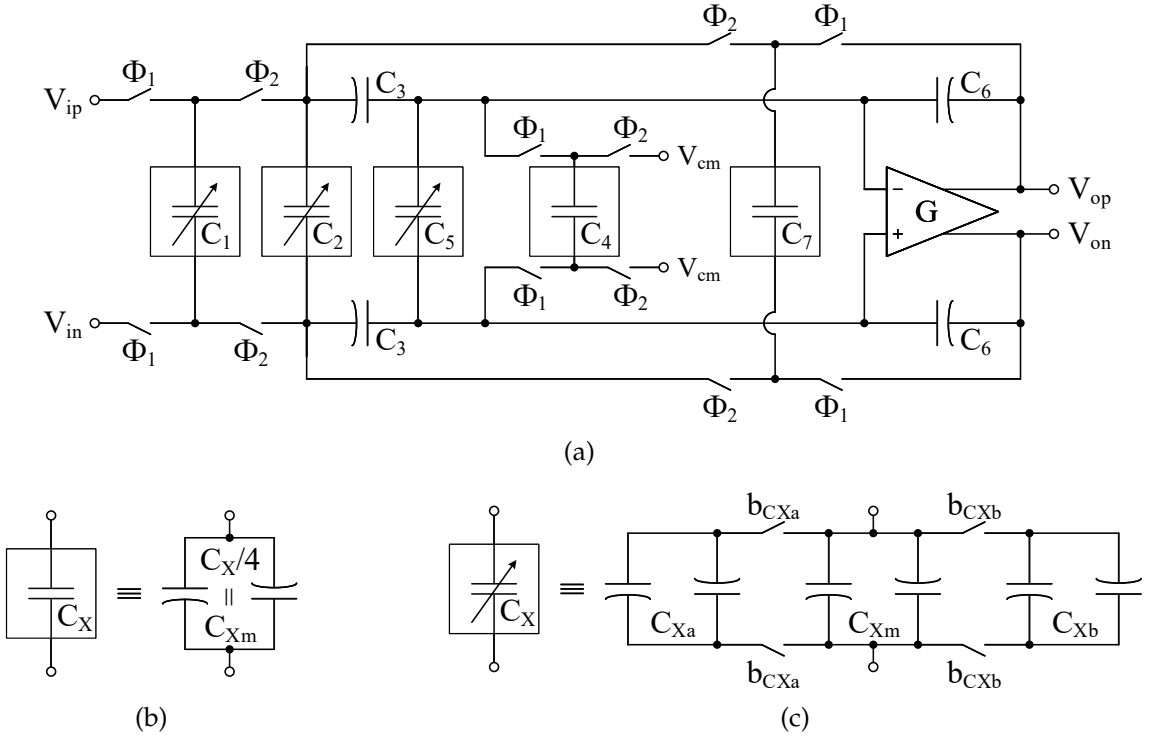


Figure 7.3: Programmable second-order bandpass SC filter using a low gain amplifier: Differential implementation of the (a) filter, (b) capacitor, and (c) programmable capacitor.

The design used in the bandpass SC filter's capacitors is shown in Table 7.1, where the capacitors were implemented using MIM capacitors.

Table 7.1: Design used in the capacitors of the bandpass SC filter.

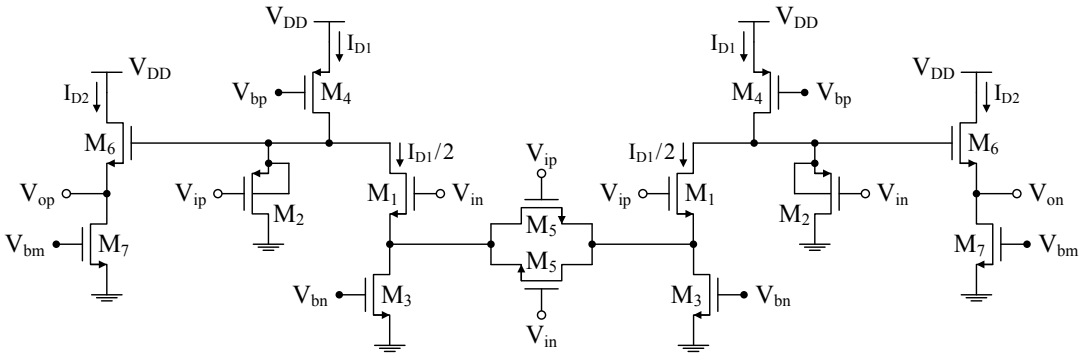
	C_{1m}	C_{1a}	C_{1b}	C_{2m}	C_{2a}	C_3	C_{4m}	C_{5m}	C_{5a}	C_{5b}	C_6	C_{7m}
Cap. (fF)	20.00	21.37	42.96	1694.26	2650.64	3341.44	28.76	110.51	174.55	299.51	149.47	268.51
Width (μm)	4.25	4.40	6.33	31.82	49.91	74.68	5.14	6.16	9.86	17.08	12.0	16.16
Length (μm)	4.25	4.40	6.33	52.64	52.64	44.38	5.14	17.08	17.08	17.08	12.0	16.16
Multiplier	1	1	1	1	1	3	1	1	1	1	1	1

The programmability of capacitors C_1, C_2, and C_5 allows a certain degree of tunability in the filter's quality factor, gain, and central frequency. The value of capacitor C_1 is controlled by bits b_{C1a} and b_{C1b}, capacitor C_2 by bit b_{C2a}, and capacitor C_5 by bits b_{C5a} and b_{C5b}.

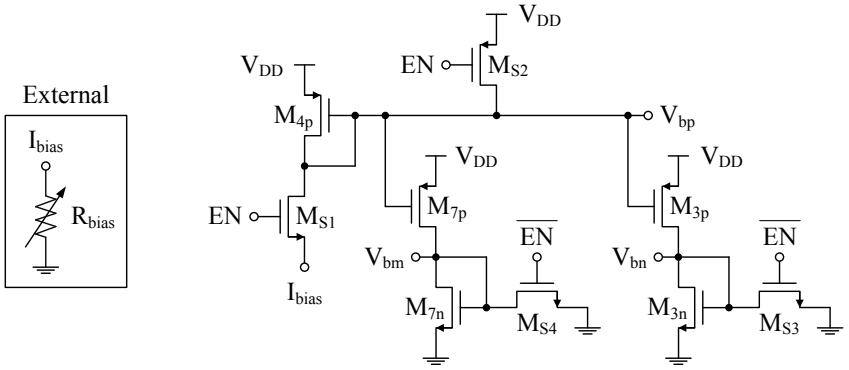
The clock phases Φ_1 and Φ_2 were generated with a two-phase non-overlapping clock generator using the architecture from Section 2.1.2.

The common-mode voltage V_{cm}, used to discharge capacitor C_4 during clock phase Φ_2, is generated using four resistances in series (W = 0.8 μm, L = 20 μm, R = 26.26 kΩ) connected between V_{DD} and V_{SS}, where the common-mode voltage is taken from the middle node. The resistance block was implemented using a serpentine layout and (floating) dummy resistors, with the same sizing, were placed at the sides to provide the same environment and protect the main resistors from mask and etching variations, resulting in a better matching between the resistors [88]. For the same reason, dummy elements were also placed at the sides of the capacitors and at the sides of the (amplifier) transistors.

The amplifier circuit was implemented using the same architecture as in Section 5.2.3, reproduced in Fig. 7.4(a) for convenience. The voltage-combiner's biasing circuit is shown in Fig. 7.4(b). To allow the amplifier to be turned off when its use is not needed, an NMOS switch (M_{S1}), driven by an enable (EN) bit, is used in series with the bias resistor (R_{bias}), minimizing the power dissipation when the bandpass filter is not needed in the acquisition channel. To ensure that the bias voltages (V_{bm}, V_{bn}, V_{bp}) are well defined when the circuit is turned off, additional switches are used to push the gates of the NMOS transistors (M_{3n} and M_{7n}) to 0 V (switches M_{S3} and M_{S4}) and of the PMOS transistor (M_{4p}) to V_{DD} (switch M_{S2}).



(a)



(b)

Figure 7.4: Voltage-combiner amplifier with source degeneration and DC level shifter: (a) differential implementation of the circuit and (b) biasing circuit.

The design used in the amplifier circuit is shown in Table 7.2, as well as some of the amplifier's metrics (gain, power, and area).

Table 7.2: Design used in the voltage-combiner amplifier and bias circuit.

Voltage-Combiner	M_1	M_2	M_3	M_4	M_5	M_6	M_7
Width (µm)	4.00	185.40	44.00	54.00	44.00	99.40	100.00
Length (µm)	0.48	0.84	1.56	0.24	1.56	0.36	1.80
# Fingers	4	18	10	30	10	28	10

Biasing Circuit	M_{S1}	M_{S2}	M_{S3}	M_{S4}	M_{3p}	M_{3n}	M_{4p}	M_{7p}	M_{7n}
Width (µm)	5.00	1.35	1.30	1.00	2.70	4.40	5.40	2.70	10.00
Length (µm)	0.36	0.36	0.36	0.36	0.24	1.56	0.24	0.24	1.80
# Fingers	1	1	1	1	2	1	3	2	1

V_{DD} (V)	V_{cm} (V)	R_{bias} (kΩ)	G (V/V)	ΔG (mV/V)	$Power_{ON}$ (µW)	$Power_{OFF}$ (nW)	Area (µm^2)
0.90	0.45	58.50	1.42	6.00	248.6	80.3	58.4 × 57.7

Since the circuit's transfer function is dependent of the gain value of the amplifier, a certain degree of tunability, mainly in the SC filter's gain around the central frequency, can be achieved by tuning the value of R_{bias}, since this resistance is implemented with a trimmer potentiometer in the printed circuit test board. The amplifier's DC gain variation as a function of R_{bias} is shown in Fig. 7.5(a). The amplifier's gain as a function of the input common-mode voltage, shown in Fig. 7.5(b), was optimized to have the minimum amount of variation in the input voltage range, to minimize the distortion introduced in the filter, as described in Section 6.1.1.1.

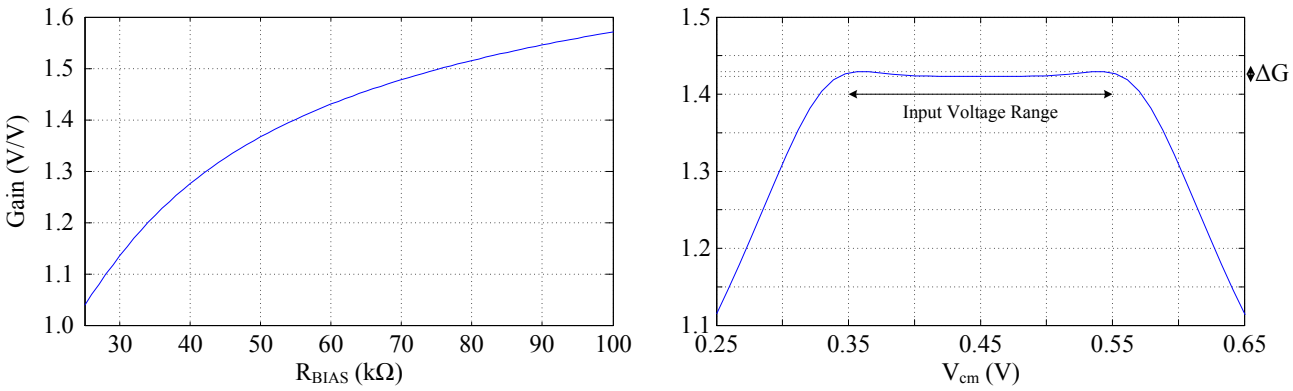


Figure 7.5: Voltage-combiner's DC gain as a function of the (a) bias resistance and (b) input common-mode voltage.

In the layout of the amplifier, which is shown in Fig. 7.6, besides the use of dummy transistors, the differential pair was implemented in common-centroid layout to further improve matching of these transistors (M_1). The layout of the complete filter, including the bonding pads, is shown in Fig. 7.7. To protect the circuit from electrostatic discharge (ESD) induced damage, two diodes were connected to each bonding pad, one connecting to V_{DD} and the other to V_{SS}.

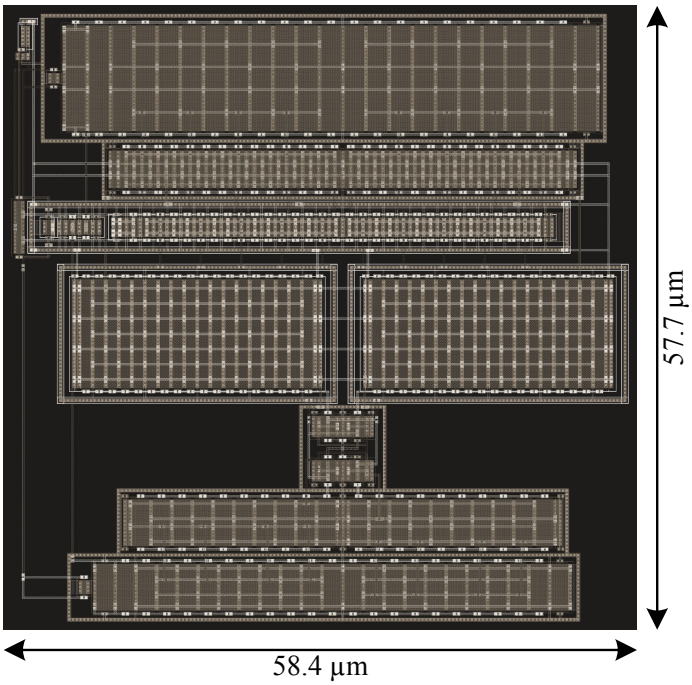


Figure 7.6: Layout of the voltage-combiner amplifier with source degeneration and DC level shifter.

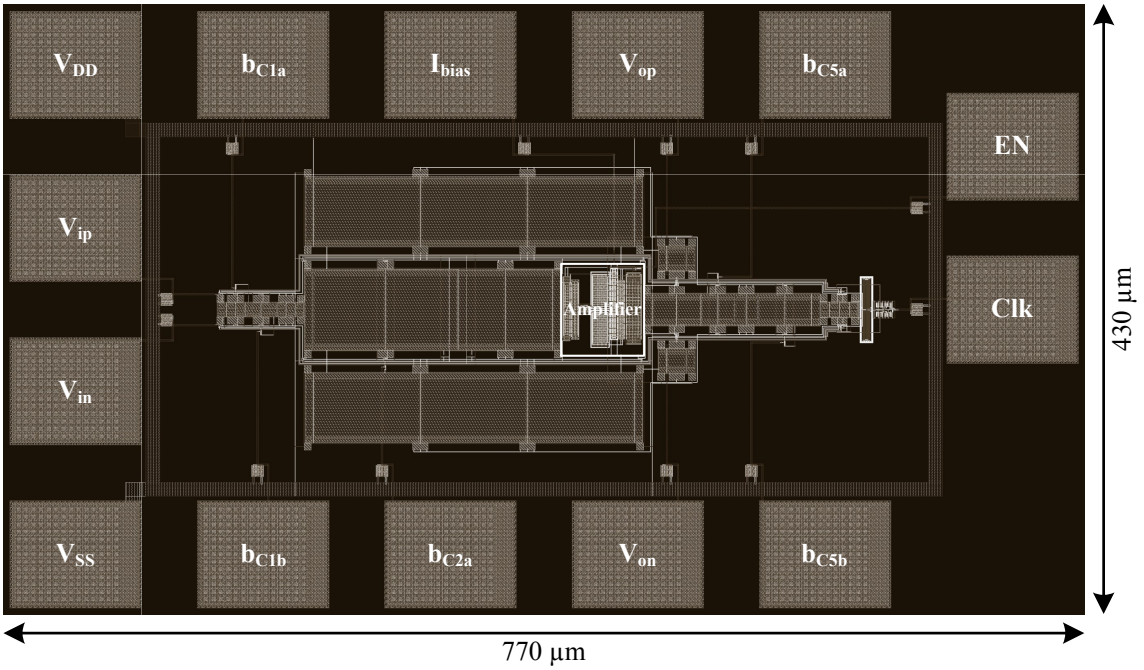


Figure 7.7: Layout of the programmable second-order bandpass SC filter.

A photo of the test board is shown in Fig. 7.8. The chip is connected to the test board by wedge bonding using gold wire. The chip's supply voltage is generated using a low dropout linear regulator with an input voltage of 3.3 V, where the regulator's output voltage is adjustable (to 0.9 V) using a trimmer potentiometer. The input signal goes through a passive DC level shifter to move its common-mode voltage to the appropriate value (0.45 V). This value is generated using a precision voltage reference circuit, with a 3.3 V input, and a trimmer potentiometer to lower the reference voltage from 1.2 V to 0.9 V.

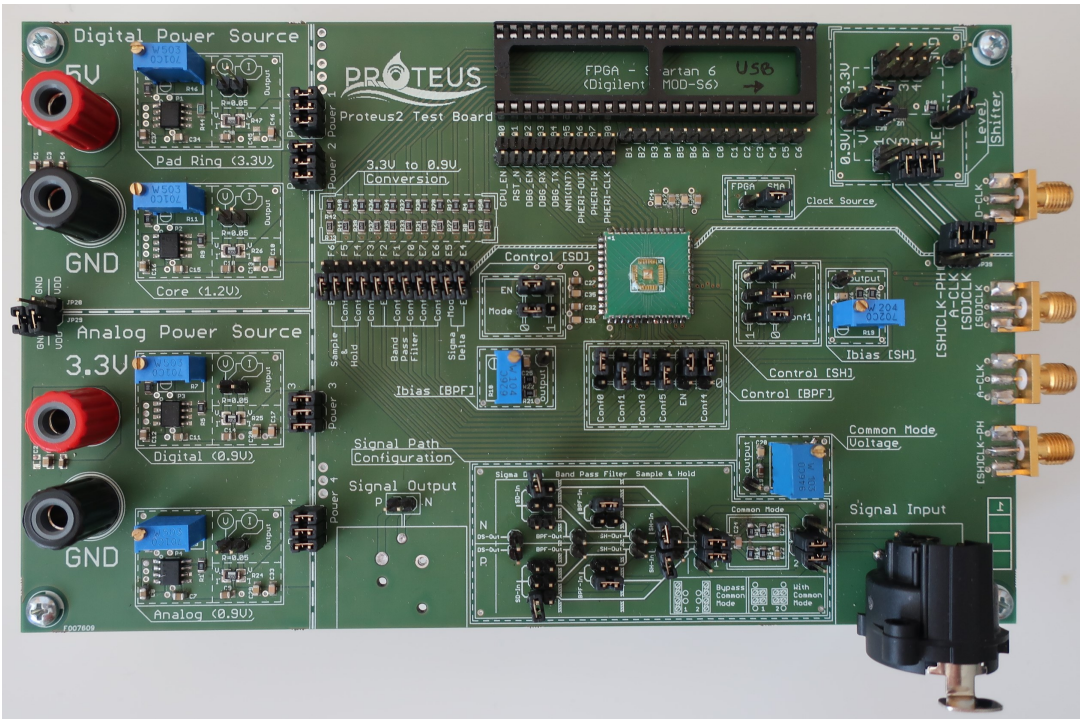


Figure 7.8: Programmable second-order bandpass SC filter test board.

A comparison between the frequency responses obtained from the schematic and post-layout (C+CC) simulations (obtained from a transient simulation of the impulse response) and from the measurement of the test board are shown in Fig. 7.9. The input signal for the test board was generated by an audio test system (Audio Precision ATS-2) with an amplitude of -30 dBV. Note that the frequency response from the test board was measured in the Audio Precision and was normalized by the signal's amplitude at the input of the SC filter, since the common-mode circuit attenuates the signal ($\approx$ 1 dBV) before it reaches the input of the bandpass SC filter. Results show that the filter's measured frequency response is in good agreement with the post-layout simulation.

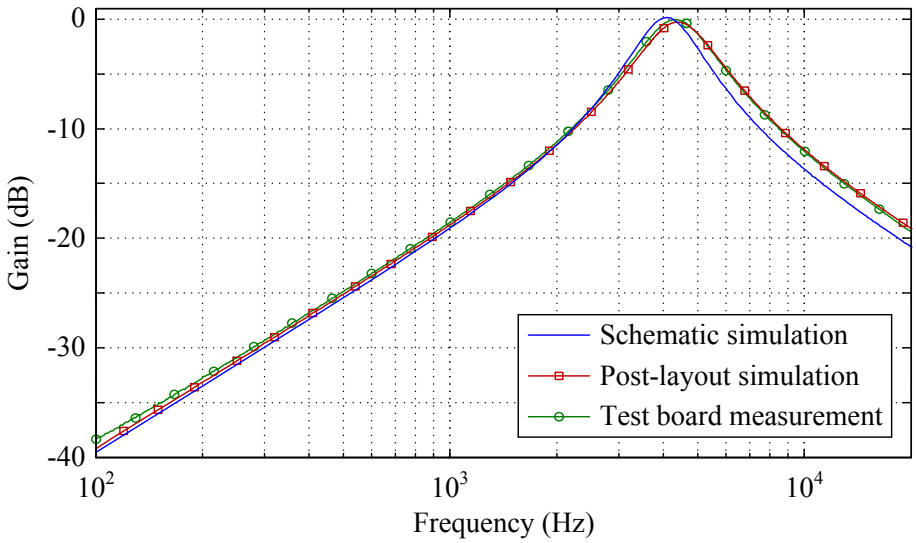


Figure 7.9: Frequency response comparison of the programmable second-order bandpass SC filter in the default bit configuration ($b_{C1a} = 1$, $b_{C1b} = 0$, $b_{C2a} = 1$, $b_{C5a} = 1$, $b_{C5b} = 0$).

The measured frequency responses for each bit configuration are shown in Fig. 7.10.

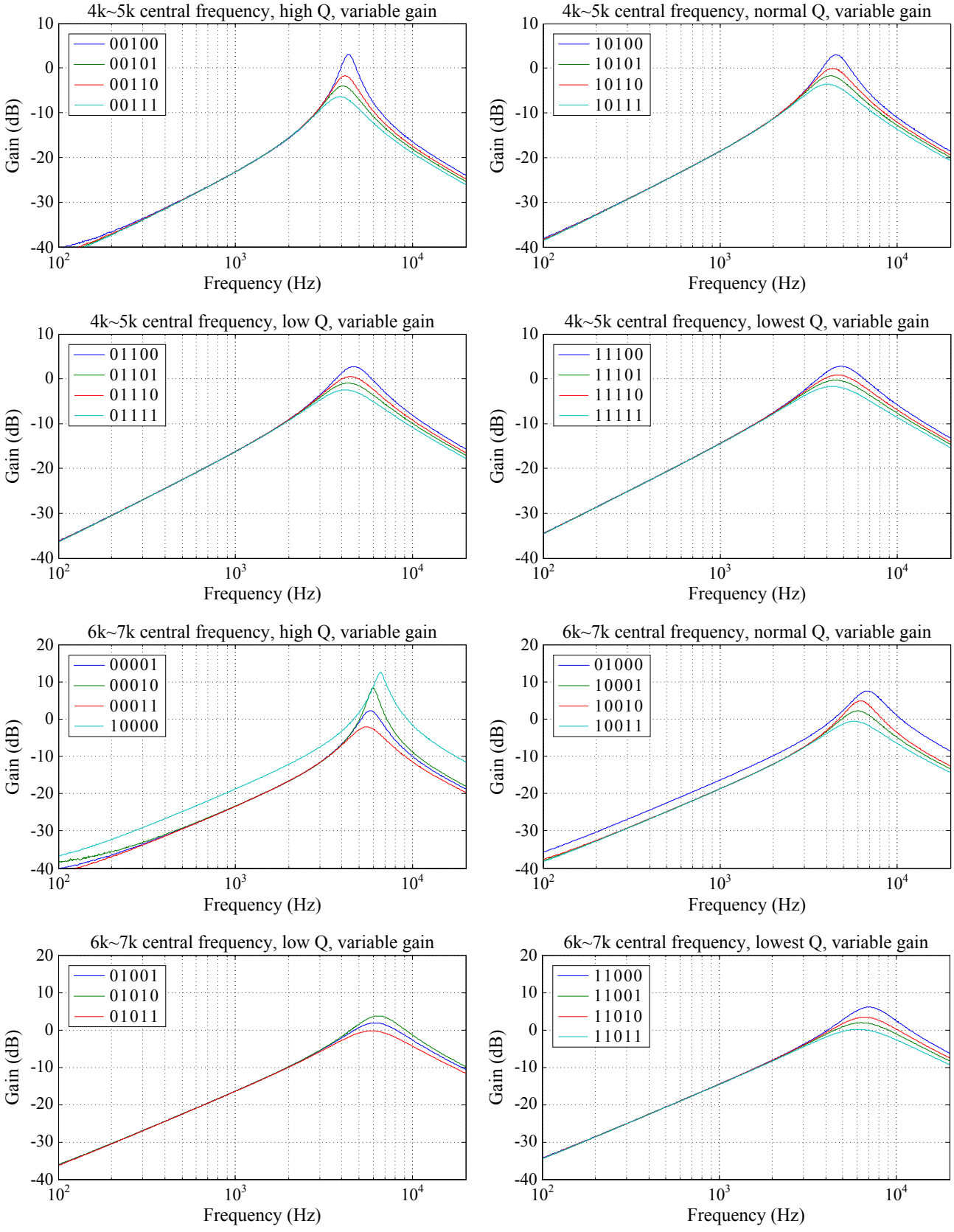


Figure 7.10: Measured frequency responses of the programmable second-order bandpass SC filter for the different bit codes (bit sequence: b_{C1a}, b_{C1b}, b_{C2a}, b_{C5a}, b_{C5b}).

A summary of the measurement results of some of the filter's metrics, for the different bit configurations, is shown in Table 7.3. Note that, due to the different gain values at the filter's central frequency, depending on the bit configuration, the measurement of the filter's THD was performed for the same signal amplitude (-30 dBV) at the output of the SC filter.

Table 7.3: Measurement summary of some of the filter metrics for the different bit configurations.

Gain (dB)	Central Frequency (kHz)	Quality Factor	THD (dB)	Power (µW)
-6.4 ∼ 12.6	3.9 ∼ 7.1	0.9 ∼ 6.9	-54.7 ∼ -61.7	248.6

The measured output spectrum of the SC filter is shown in Fig. 7.11, for the default bit configuration, where the input signal has two tones (f_1 = 4.1 kHz and f_2 = 4.8 kHz) and an amplitude of -30 dBV, resulting in a 3rd order intermodulation product (IM3) of -66.5 dB.

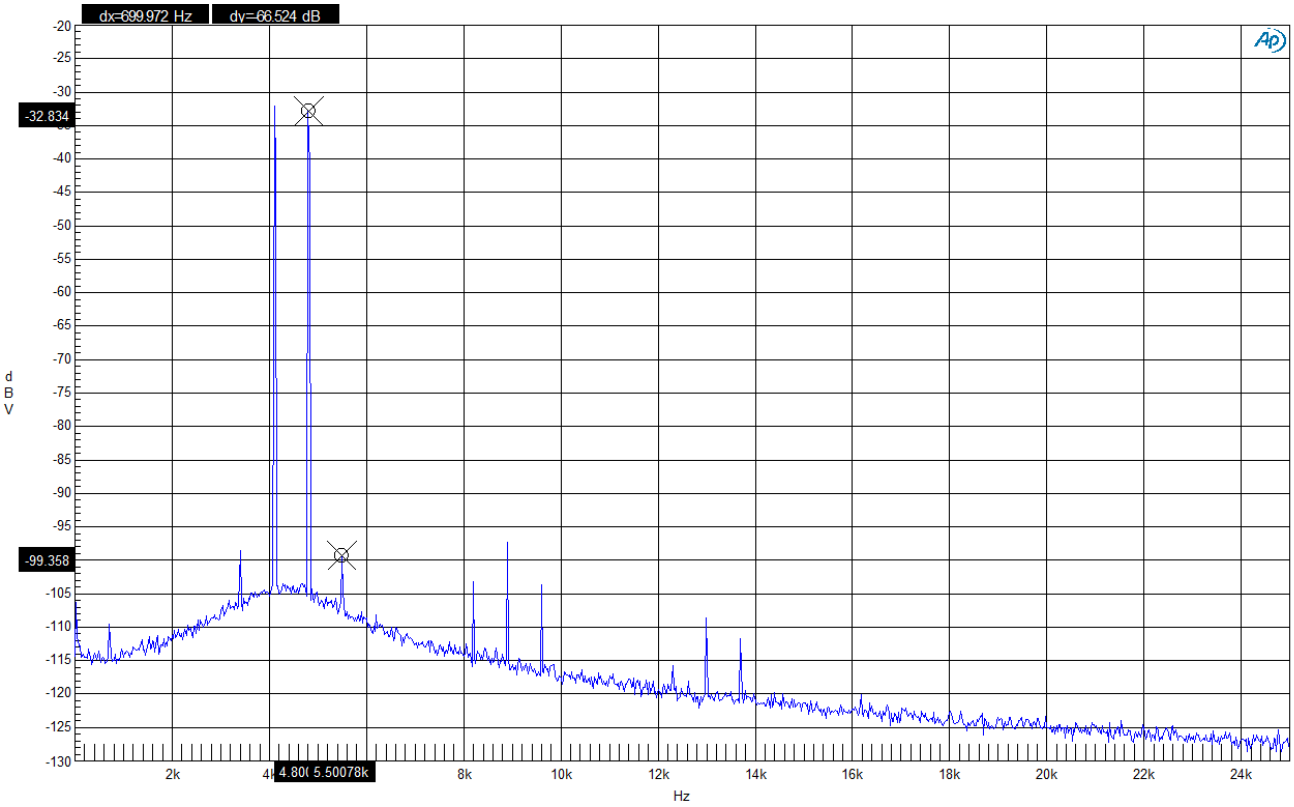


Figure 7.11: Measured output spectrum of the programmable second-order bandpass SC filter in the default bit configuration for an input with two tones (f_1 = 4.1 kHz and f_2 = 4.8 kHz).

7.2 50 Hz Notch Switched-Capacitor Filter

The large ratio between the sampling frequency (F_s = 1 MHz) and the notch frequency (f_p = 50 Hz) of the notch filter results in a very large dispersion between the capacitors' values in a SC circuit. In order to decrease the dispersion, it is required to use SC structures capable of delivering a small charge using a larger sized capacitor. To achieve this objective, the concept of a charge division branch (CDB) is used, which consists in destroying part of the charge stored in a switched network's capacitor before transferring it to the rest of the circuit (transfer phase). If this network is connected to a virtual ground node during the transfer phase, the circuit will effectively see a smaller capacitor than its actual value, due to the loss of part of the charge. This concept is demonstrated with the parasitic-sensitive SC integrator [20] described in Section 2.2.1.

7.2.1 Parasitic-Sensitive Switched-Capacitor Integrator using a Charge Division Branch

The SC integrator shown in Fig. 7.12 is used to demonstrate how to increase the capacitance value in a switched network using a CDB, while producing, from an ideal standpoint, the same transfer function.

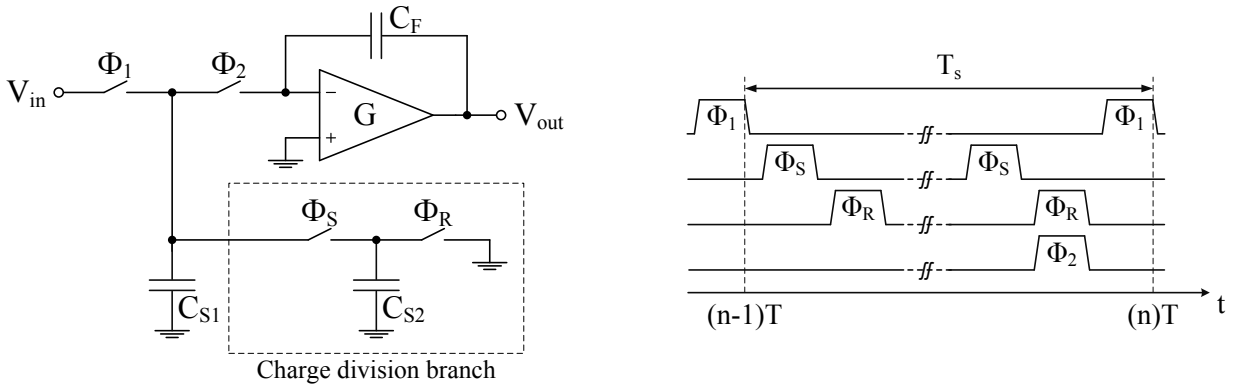


Figure 7.12: Parasitic-sensitive switched-capacitor integrator using charge division branch.

Neglecting the CDB and the effect of the parasitic capacitances, using the principle of charge conservation, the circuit's transfer function is given by (7.1), where the amplifier's gain is considered as infinity.

$$H^{\Phi_1}(z) = \frac{C_{S1}}{C_F}\left(\frac{1}{z-1}\right) \tag{7.1}$$

Considering now the CDB and the phase diagram shown in Fig. 7.12, capacitor C_{S1} will lose part of its charge after each share(Φ_S)/reset(Φ_R) cycle. If capacitors C_{S1} and C_{S2} have the same value, with each cycle, C_{S1} will lose half of its charge. If this process is repeated during k share/reset cycles, the transfer function (7.2) is obtained.

$$H^{\Phi_1}(z) = \frac{{C_{S1}}^{(k+1)}}{C_F(C_{S1}+C_{S2})^k}\left(\frac{1}{z-1}\right)\Bigg|_{C_S=C_{S1}=C_{S2}} = \frac{1}{2^k}\frac{C_S}{C_F}\left(\frac{1}{z-1}\right) \tag{7.2}$$

Equation (7.2) shows that, in order to obtain the same transfer function as in (7.1), capacitor C_S has to be 2^k times larger than C_{S1}. Using this approach it is possible to reduce the ratio of C_F/C_S, which becomes very large when the ratio F_s/f_p increases well beyond the typical $10-100$ value.

Analyzing the circuit without the CDB, the charge equations are given by:

$$Q_{C_{S1}}^{\Phi_1} + Q_{C_F}^{\Phi_1} = Q_{C_{S1}}^{\Phi_2} + Q_{C_F}^{\Phi_2} \Leftrightarrow V_{in}^{\Phi_1} C_{S1} + (G^{-1} - 1) V_{out}^{\Phi_1} C_F = G^{-1} V_{out}^{\Phi_2} (C_{S1} + C_F) - V_{out}^{\Phi_2} C_F \quad (7.3)$$

and, using the CDB with one share/reset cycle, the charge equations are given by:

$$\begin{aligned} Q_{C_{S1}}^{\Phi_R} + Q_{C_F}^{\Phi_R} = Q_{C_{S1}}^{\Phi_2} + Q_{C_F}^{\Phi_2} &\Leftrightarrow V_{C_{S1}}^{\Phi_S} C_{S1} + (G^{-1} - 1) V_{out}^{\Phi_S} C_F = G^{-1} V_{out}^{\Phi_2} (C_{S1} + C_F) - V_{out}^{\Phi_2} C_F \Leftrightarrow \\ &\Leftrightarrow V_{in}^{\Phi_1} \frac{C_{S1}}{C_{S1} + C_{S2}} C_{S1} + (G^{-1} - 1) V_{out}^{\Phi_1} C_F = G^{-1} V_{out}^{\Phi_2} (C_{S1} + C_F) - V_{out}^{\Phi_2} C_F \Rightarrow \quad (7.4) \\ &\xRightarrow[C_{S1} = C_{S2} = C_S]{} V_{in}^{\Phi_1} \frac{C_S}{2} + (G^{-1} - 1) V_{out}^{\Phi_1} C_F = G^{-1} V_{out}^{\Phi_2} (C_S + C_F) - V_{out}^{\Phi_2} C_F \end{aligned}$$

Equating the results from (7.3) and (7.4), looking only at the terms that depend on C_{S1}, C_{S2}, and C_S:

$$V_{in}^{\Phi_1} C_{S1} - G^{-1} V_{out}^{\Phi_2} C_{S1} = V_{in}^{\Phi_1} \frac{C_S}{2} - G^{-1} V_{out}^{\Phi_2} C_S \quad (7.5)$$

This result shows that, in order for this approach to work, i.e, for equation (7.5) to be valid, capacitor C_S needs to be 2 times larger than C_{S1} and the gain of the amplifier needs to be high to minimize the memory effect from C_S (C_{S1}) during the transfer phase, i.e., this approach can be used to increase the value of a capacitor in a switched network when, during the transfer phase, the switched network connects to a virtual ground node.

Note that, when using CDBs, the circuit becomes sensitive to the effects of parasitic capacitances, even if the switched branch is implemented with a π network. For this reason, it is better to implement the switched branches with T networks, as shown Fig. 7.12, to minimize the amount of switches used and, as a consequence, reduce the amount of parasitic capacitances added to the circuit when compared with π networks.

The use of CDBs is best suited for low frequency applications since, for each share/reset cycle, the number of phases used increases by two, where the phases length is given by:

$$t_{ph} = \frac{T_s}{2 \times (1 + k)} \quad (7.6)$$

Depending on the size of the capacitors and the number of share/reset cycles in a sampling period, the amplifier's GBW will have to increased.

7.2.2 Notch Switched-Capacitor Filter using Charge Division Branches

The notch SC filter, used to remove the 50 Hz interference from the signal's path, is shown in Fig.7.13(a) and is based on the Fleischer-Laker architecture [73]. The switches were implemented using asymmetric transmission gates to reduce the switches' resistance non-linearity and to have approximately the same resistance during normal operation (sinusoidal input) on both sides of the differential circuit.

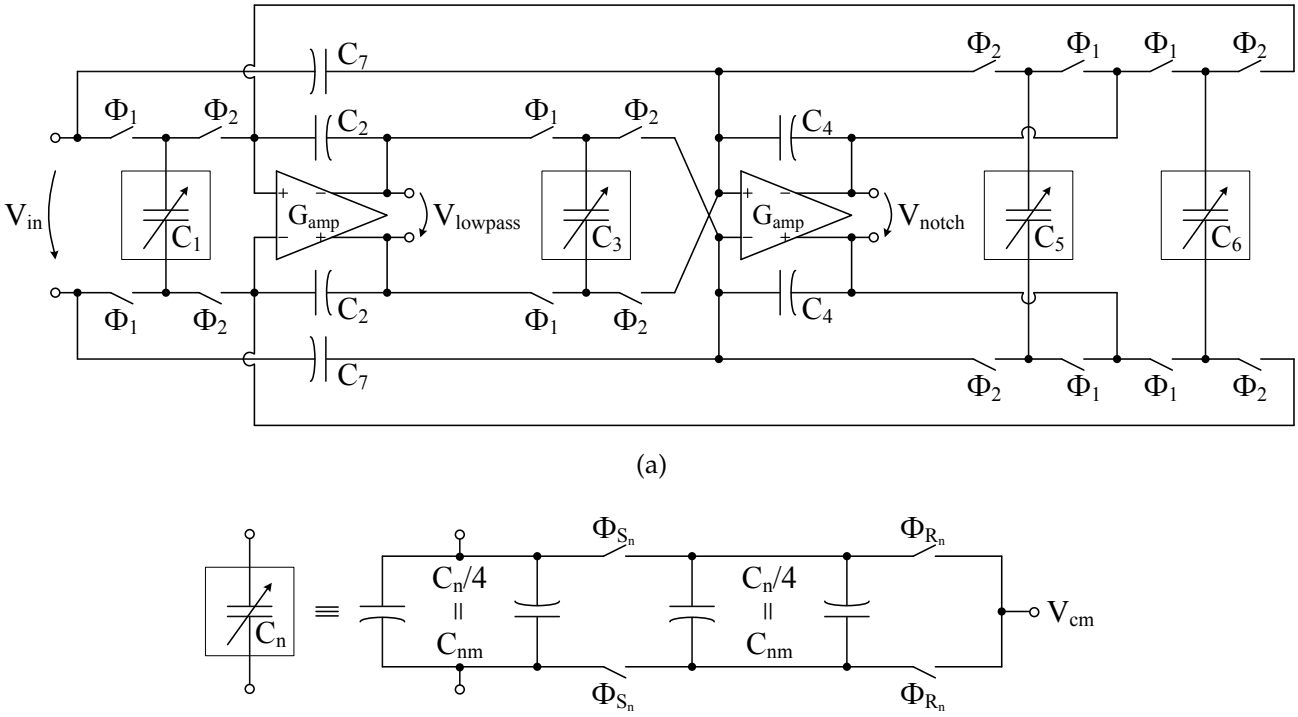


Figure 7.13: Notch SC filter: Differential implementation of the (a) filter and (b) capacitor using charge division branch, for n = 1, 3, 5, and 6.

Considering the equivalent circuits during phase Φ_1 and phase Φ_2, shown in Fig. 7.14, which neglects the CDBs, using the principle of charge conservation and considering that the output is sampled at the end of clock phase Φ_2, the system of charge equations is (7.7) obtained.

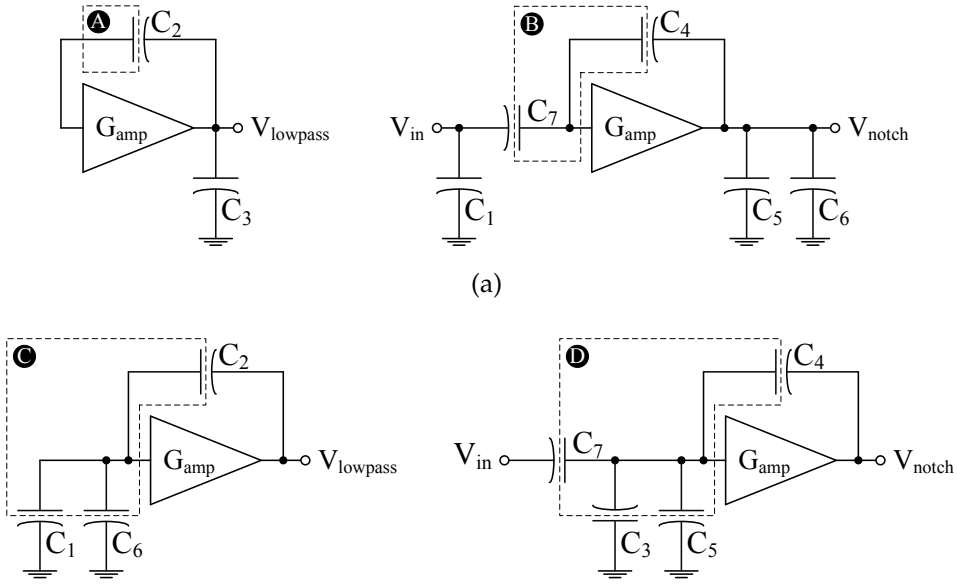


Figure 7.14: Equivalent circuit during (a) phase Φ_1 and (a) phase Φ_2.

$$
\begin{aligned}
&\text{(A)}\ V_{lowpass}[n-1]\left(\frac{C_2}{G_1}-C_2\right) = V_{lowpass}[n-1/2]\left(\frac{C_2}{G_1}-C_2\right)\\
&\text{(B)}\ -V_{in}[n-1]\,C_7 + V_{notch}[n-1]\left(\frac{C_4+C_7}{G_2}-C_4\right) = -V_{in}[n-1/2]\,C_7 + V_{notch}[n-1/2]\left(\frac{C_4+C_7}{G_2}-C_4\right)\\
&\text{(C)}\ V_{in}[n-1/2]\,C_1 + V_{lowpass}[n-1/2]\left(\frac{C_2}{G_1}-C_2\right) + V_{notch}[n-1/2]\,C_6 = V_{lowpass}[n]\left(\frac{C_1+C_2+C_6}{G_1}-C_2\right) \quad (7.7)\\
&\text{(D)}\ -V_{in}[n-1/2]\,C_7 - V_{lowpass}[n-1/2]\,C_3 + V_{notch}[n-1/2]\left(\frac{C_4+C_7}{G_2}-C_4+C_5\right) =\\
&\quad = -V_{in}[n]\,C_7 + V_{notch}[n]\left(\frac{C_3+C_4+C_5+C_7}{G_2}-C_4\right)
\end{aligned}
$$

Solving the system of charge equations (7.7) and considering the gain of the amplifiers as infinity, using the $\mathcal{Z}$-transform the discrete-time transfer function (7.8) is obtained. Note that the effect of the notch comes from the cross-coupling of the switched network formed by capacitor C_3 at the input of the second amplifier, allowing to achieve complex conjugate zeros on the unit circle.

$$
H^{\Phi_2}_{notch}(z) = -\frac{C_3C_6C_7 + C_2C_7\,(C_4 - C_5 + C_4\,z^{-1/2})(z^{-1/2}+1)(z^{-1/2}-1)^2 + C_3\,(-C_1C_4 + C_6C_7)\,z^{-1/2}}{C_3C_4C_6 + C_2C_4\,\big(C_5 + C_4\,(z-1)\big)\,(z-1)} \quad (7.8)
$$

Using the ideal transfer function (7.8), the SC filter was manually designed using the specifications shown in Table 7.4. This was done to get a general idea on the capacitor values dispersion and the number of share/reset cycles necessary increase the value of the smaller capacitors to more reasonable values, i.e., above the minimum value allowed by the technology. A possible design for the notch SC filter from an ideal standpoint is shown in Table 7.5.

Table 7.4: Frequency response specifications used in the design of the notch SC filter.

Frequency Specifications	$F_{passlow}$ (Hz)	F_{notch} (Hz)	F_{sensor} (kHz)	F_s (MHz)
	0.5	50.0	5.0	1.0
Gain Specifications (dB)	≈ 0	$-\infty$	≈ 0	≈ 0

Table 7.5: Possible design for the ideal notch SC filter (C_{max}/C_{min} = 3333.33).

C_1 (fF)	C_2 (fF)	C_3 (fF)	C_4 (fF)	C_5 (fF)	C_6 (fF)	C_7 (fF)	G_1	G_2	F_s (MHz)
0.79	10000.00	0.79	10000.00	20.00	0.75	10000.00	-∞	-∞	1.00

From these results, considering the approach shown in Fig.7.13(b) for the implementation of the charge division branches and that the minimum allowed value for the capacitors, in the technology used, is 17 fF, a maximum number of 7 share/reset cycles are used to increase the values of capacitors C_1, C_3, C_5, and C_6. The exact number of cycles used for each capacitor will be individually controlled by 2 bits to allow a certain degree of tunability in the notch's frequency, quality factor, and low frequency gain. In total, the notch filter is controlled by 12 phases: one phase for the sampling of the signal (Φ_1); four share (Φ_{S_n}) and four reset phases (Φ_{R_n}) to allow a higher capacitance values in capacitors C_1, C_3, C_5, and C_6; one phase for transferring (Φ_2); and two phases (Φ_{A1} and Φ_{A2}) to control the amplifier's switched-capacitor common-mode feedback (SC-CMFB) circuit.

The diagram of the phases controlling the notch SC filter and the charge division branches is shown in Fig. 7.15.

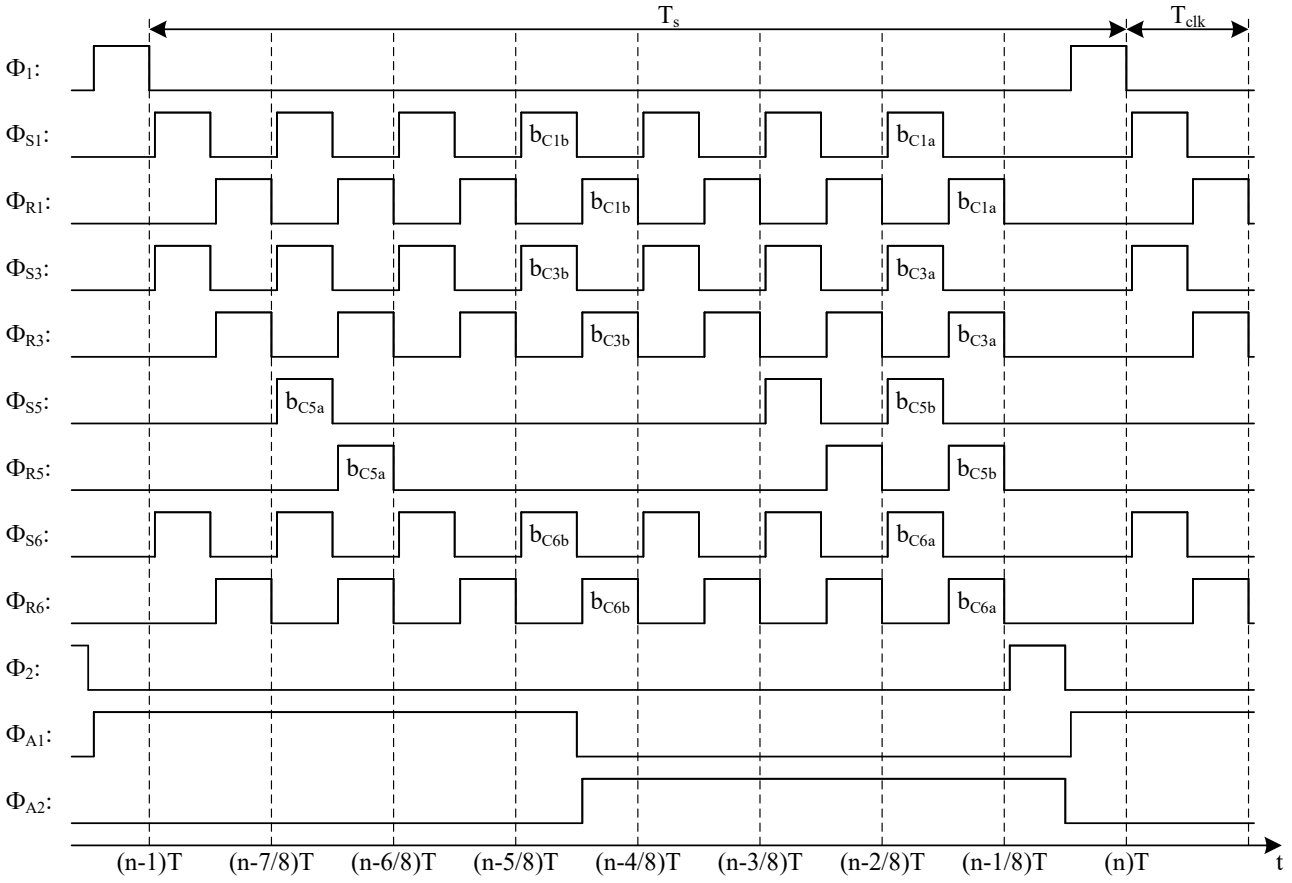


Figure 7.15: Diagram of the phases controlling the notch SC filter and the charge division branches.

where, the number of share/reset cycles in phases $\Phi_{S1/R1}$ (C_1), $\Phi_{S3/R3}$ (C_3), and $\Phi_{S6/R6}$ (C_6) is variable between 5 and 7, and is controlled by bits $b_{C1a/C1b}$, $b_{C3a/C3b}$, and $b_{C6a/C6b}$, respectively. The number of share/reset cycles in phase $\Phi_{S5/R5}$ (C_5) is variable between 1 and 3, and is controlled by bits $b_{C5a/C5b}$.

Based on the default number of share/reset cycles used in each CDB, the final filter design, shown in Table 7.6, was obtained. Note that the capacitance values in the CDBs did not increase by the expected factor of 2^k in relation to the design shown in Table 7.5, where k is the number of share/reset cycles. This is due to the parasitic capacitances from the switches and from the capacitors top and bottom plates, that have to be compensated into the main capacitors, and due to the finite gain of the amplifiers, reducing the capacitance values from their expected value.

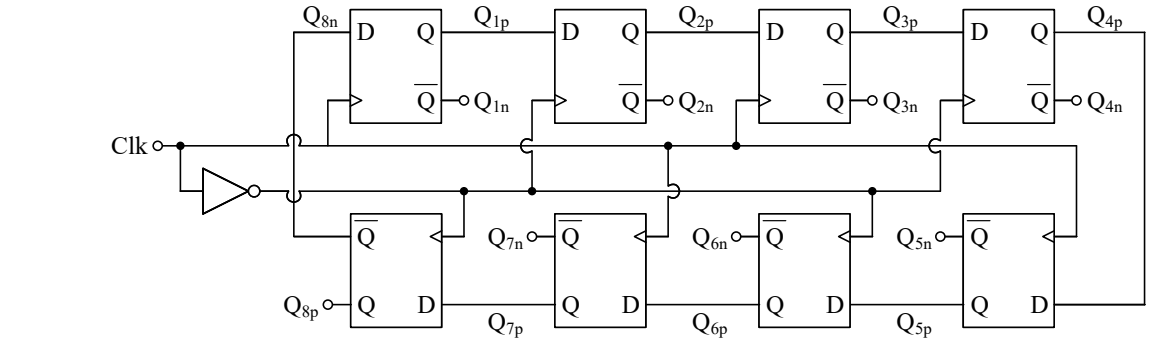
Table 7.6: Capacitor design and default number of share/reset cycles used in the notch SC filter (C_{max}/C_{min} = 140.37).

	C_{1m}	C_2	C_{3m}	C_4	C_{5m}	C_{6m}	C_7
Capacitance (fF)	35.57	10005.55	35.57	10005.55	19.56	17.82	10005.55
Width (μm)	5.74	99.80	5.74	99.80	4.20	4.00	99.80
Length (μm)	5.74	99.80	5.74	99.80	4.20	4.00	99.80
Default no. of cycles	6	-	6	-	1	7	-

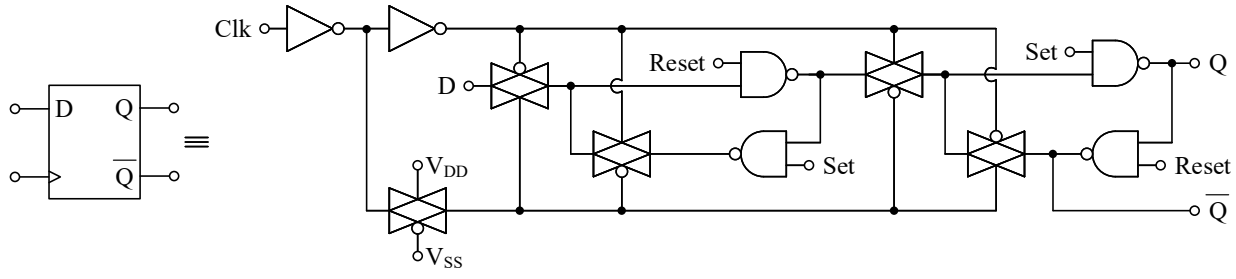
7.2.2.1 Clock phase generator

The phases necessary to control the notch SC filter and the charge division branches, shown in Fig. 7.15, are generated using the acquisition channel's master clock (F_{clk} = 8 MHz) and the divide by 8 clock circuit shown in Fig. 7.16(a). For simplicity reasons, the Set and Reset signals (both active-low) are not shown in the figure. The Reset signal is used to ensure the correct operation of the divide by 8 circuit, in case any of the flip-flops initiates in the wrong state.

The flip-flops use both the clock and the inverted clock signal, i.e, the odd numbered flip-flops use the clock signal and the even numbered ones use the inverted clock, allowing the generation of 16 phases with a period of 1 µs from a 8 MHz clock signal. These phases are obtained by combining the appropriate flip-flop outputs (Q_{1p-8p}, Q_{1n-8n}) using AND gates. One of the inputs of these AND gates uses two inverters in series with a large length (L = 1.20 µm) to increase the delay time between the phases and ensure that they do not overlap. Note that, for this to happen, the flip-flops' output fed to this input has to be the one controlling the rising edge of the new phase. The circuit used to implement the flip-flop D is shown in Fig. 7.16(b).



(a)



(b)

Figure 7.16: Circuit implementation of the (a) divide by 8 clock circuit and (b) flip-flop D.

The two phases used in the amplifier's common-mode feedback (CMFB) circuit are obtained from the two-phase non-overlapping clock generator from Section 2.1.2, using as input the output of the first flip-flop (Q_{1p}). Each phase has a buffer at the output so they are able to drive the switches in the notch SC filter. This buffer is implemented with two inverters in series with minimum length (L = 120 nm).

7.2.2.2 Folded-cascode amplifier

The amplifier circuits used in the notch SC filter were implemented using the folded-cascode amplifier, shown in Fig. 7.17. The SC-CMFB circuit is implemented using the conventional architecture described in [89]. The bias current is generated using a series of six resistors, where the two closer to the supply voltage are in parallel with PMOS switches driven by two control bits. This allows a programmability of $\pm$ 20 % in the desired value of the bias current. Note that due to the limited number of available pads, both amplifier's in the SC filter use the same control bits.

To allow the notch SC filter to be turned off when its use is not needed, an NMOS switch, driven by an enable (*EN*) bit, is used in series with the bias resistor (R_{bias}). To ensure that the bias voltages (V_{bn}, V_{bp}, V_{casn}, V_{casp}) are well defined when the circuit is turned off, additional switches are used to push the gates of the NMOS transistors (M_{b2} and M_{b3}) to 0 V (switches M_{S2} and M_{S3}) and of the PMOS transistors (M_{b0} and M_{b1}) to V_{DD} (switches M_{S4} and M_{S5}).

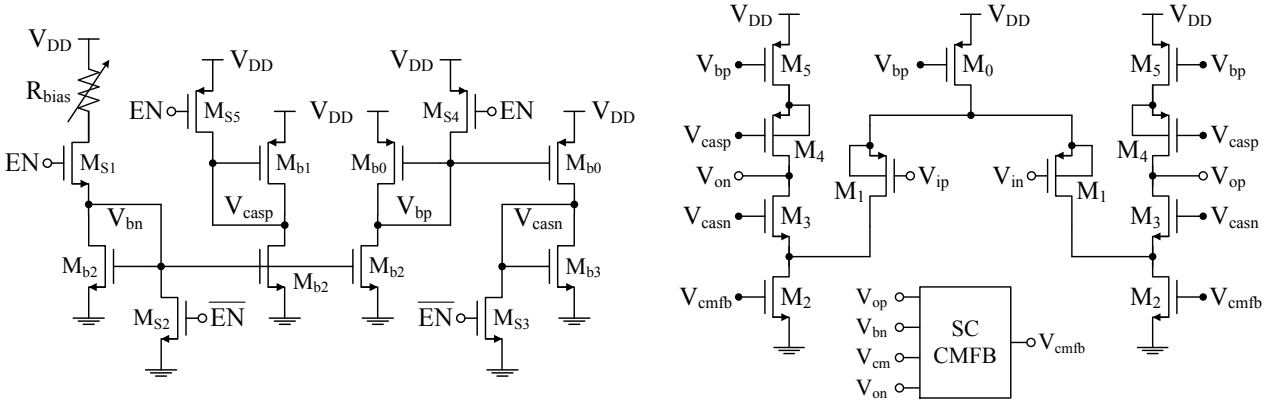


Figure 7.17: Folded-cascode amplifier: (a) Bias circuit, (b) Differential implementation.

The folded-cascode amplifiers were optimized based on the time-domain methodology from Section 5.1.1.2. The design used in the amplifiers is shown in Table 7.7, as well as some of the amplifier's metrics (gain, power, and area). The common-mode voltage V_{cm}, used in the CMFB circuit (and in the CDBs), is generated using six resistances in series (W = 0.8 µm, L = 14.39 µm, R = 18.72 kΩ) connected between V_{DD} and V_{SS}, where the common-mode voltage is taken from the middle node.

Table 7.7: Design used in the folded-cascode amplifiers.

Folded-Cascode	M_0	M_1	M_2	M_3	M_4	M_5
Width (µm)	88.00	84.00	100.00	60.00	64.00	26.40
Length (µm)	0.64	0.20	0.72	1.70	0.64	0.64
# Fingers	20	14	20	8	10	6

Biasing Circuit	M_{S1}	M_{S2}	M_{S3}	M_{S4}	M_{S5}	M_{b0}	M_{b1}	M_{b2}	M_{b3}
Width (µm)	10.00	2.50	1.00	2.20	0.72	8.80	0.72	10.00	1.00
Length (µm)	0.36	0.72	0.72	0.64	0.36	0.64	0.40	0.72	0.72
# Fingers	4	1	1	1	2	4	1	4	1

V_{DD} (V)	V_{cm} (V)	R_{bias} (kΩ)	G_{amp} (dB)	$Power_{ON}$ (µW)	$Power_{OFF}$ (nW)	Area (µm^2)
0.90	0.45	95.06 $\pm$ 20 %	66.0	131.2	37.7	73.2 $\times$ 75.0

7.2.2.3 Layout and simulation results of the complete notch SC filter

The layout of the folded-cascode amplifiers and of the clock phase generator is shown in Fig. 7.18 and Fig. 7.19, respectively.

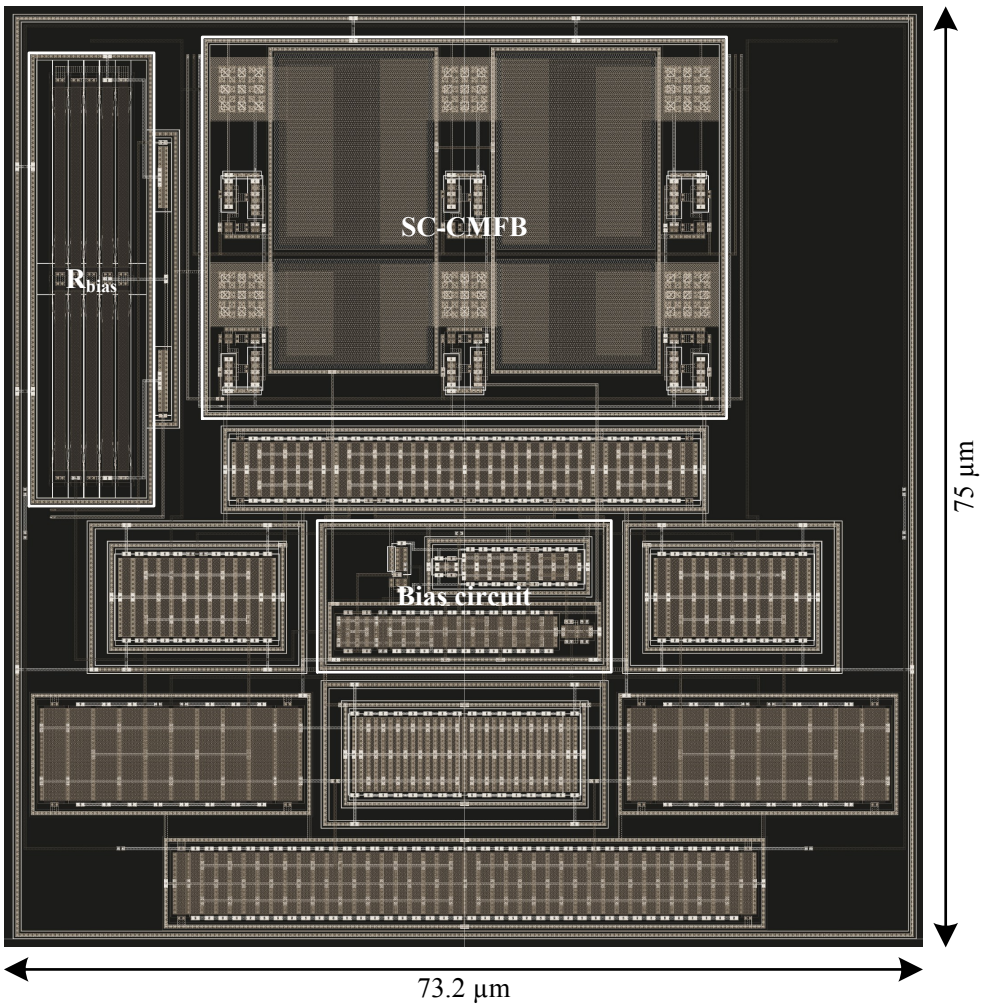


Figure 7.18: Layout of the folded-cascode amplifier and common-mode feedback circuit.

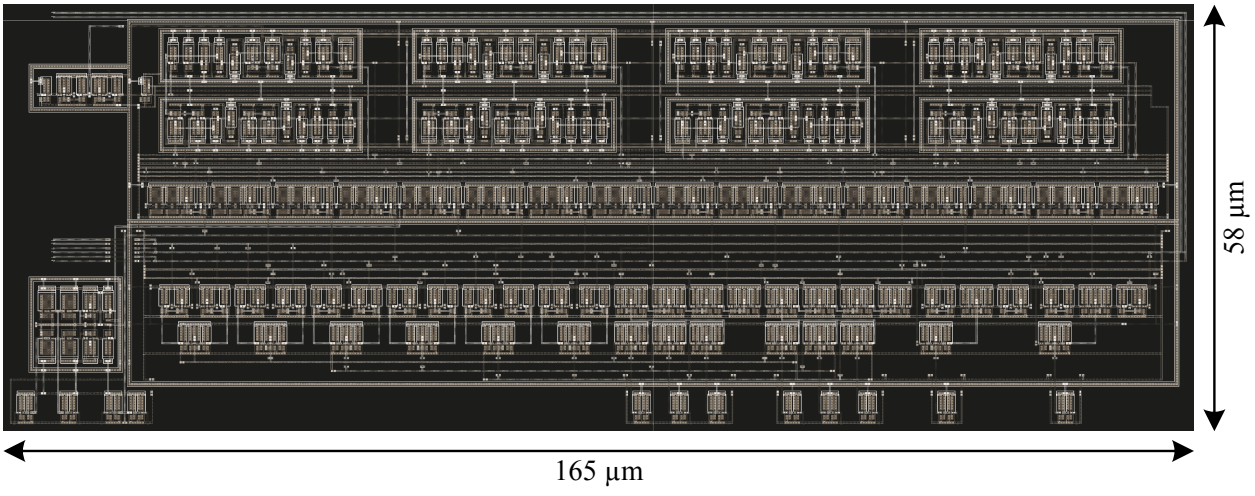


Figure 7.19: Layout of the clock phase generator circuit.

The layout of the complete notch SC filter is shown in Fig. 7.20. To improve matching, dummy elements were placed on the edges of each device (transistors, capacitors, resistors). The switches were placed on the outside part of the circuit, to keep the clocks phases as far away as possible from the switched networks' capacitors. This was done because, after sampling the signal, these capacitors discard most of the charge they store, until only a small amount is left, making them very susceptible to mismatches and interferences from the clock signals.

To minimize the coupling (interference) between the clock phases, the metals containing these signals alternate between metal levels, with one level of interval (for example, $m2$, $m4$, $m2$, $m4$, etc.), and with metal $m1$, connected to ground, below these metals to have most of the coupling to ground instead of to the other phases. Since these metals are very long, and to avoid antenna problems, at every ≈ 80 µm the metals change level, i.e, what was metal $m2$ changes to $m4$ and what was metal $m4$ changes to $m2$.

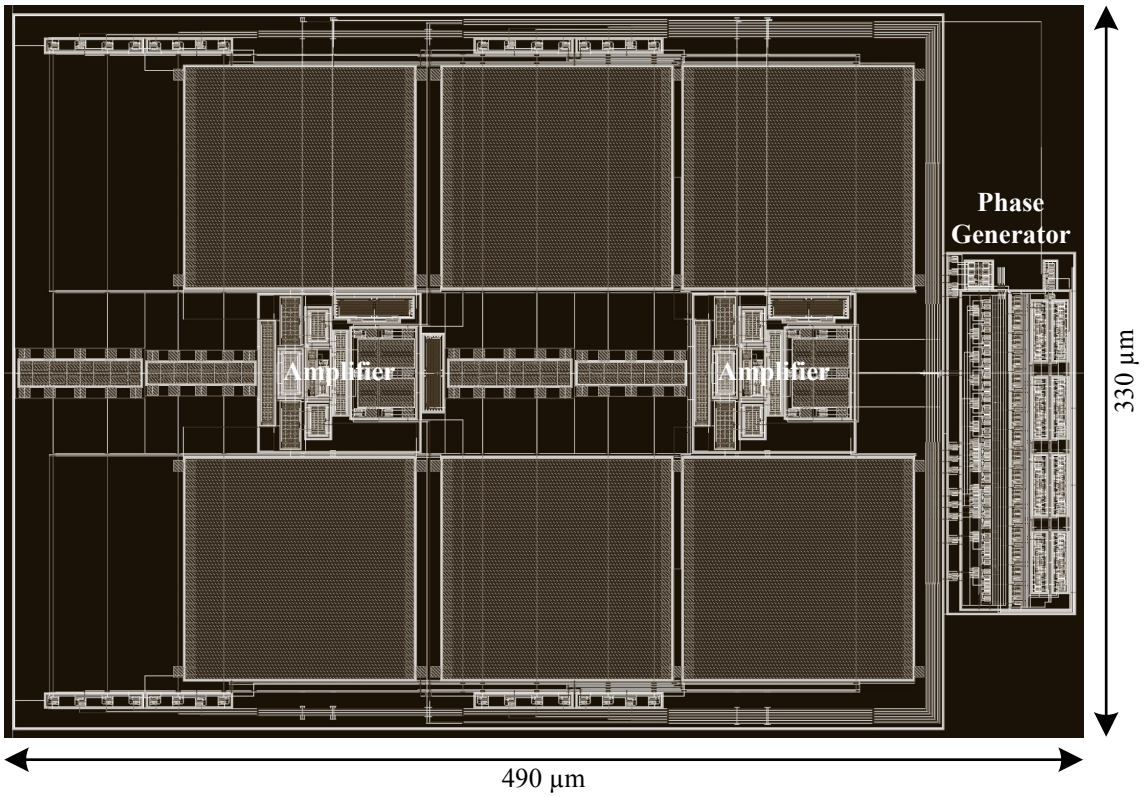


Figure 7.20: Layout of the notch SC filter using charge division branches.

Since the ratio between F_s/f_p is very large, the length of the transient simulation necessary to accurately calculate the filter's frequency response, i.e., to reach as far back as 50 Hz from 1 MHz, is very long. Considering the simulation of the extracted view (R+C+CC), this process would take several days. For this reason, in this case, the comparison between the frequency responses from the schematic and post-layout (R+C+CC) simulations, shown in Fig. 7.21, were obtained from a periodic steady state (PSS)/periodic alternating current (PAC) simulation, that takes substantially less time (few hours).

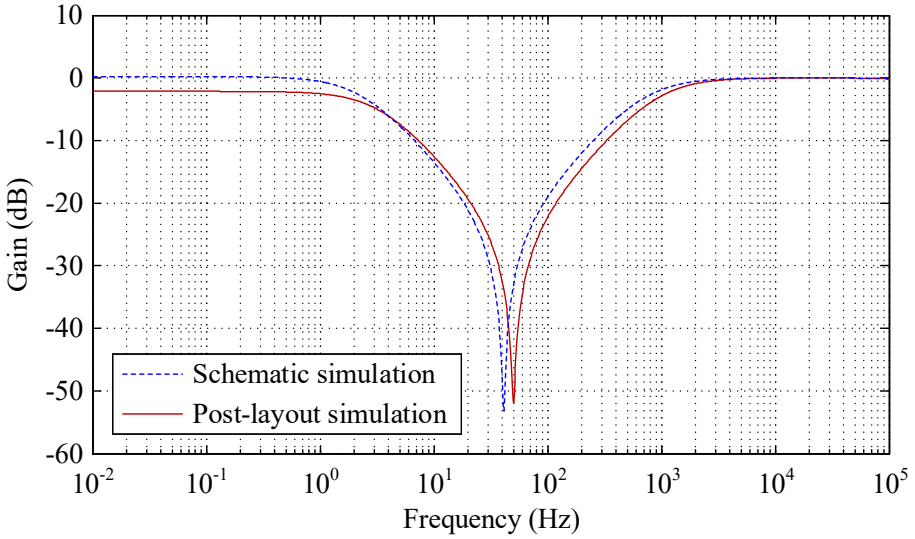


Figure 7.21: Frequency response comparison of the notch SC filter for the default number of share/reset cycles ($\Phi_{S1,R1} = 6$, $\Phi_{S3,R3} = 6$, $\Phi_{S5,R5} = 1$, $\Phi_{S6,R6} = 7$).

The notch SC filter's frequency responses are shown in Fig. 7.22, changing the number of share/reset cycles of a single capacitor at a time, from the default configuration. The notch's frequency is controlled by the number of share/reset cycles in phases $\Phi_{S1,R1}$ and $\Phi_{S3,R3}$; the quality factor by the number of cycles in phases $\Phi_{S5,R5}$; and the low frequency gain by phases $\Phi_{S1,R1}$, $\Phi_{S3,R3}$, and $\Phi_{S6,R6}$.

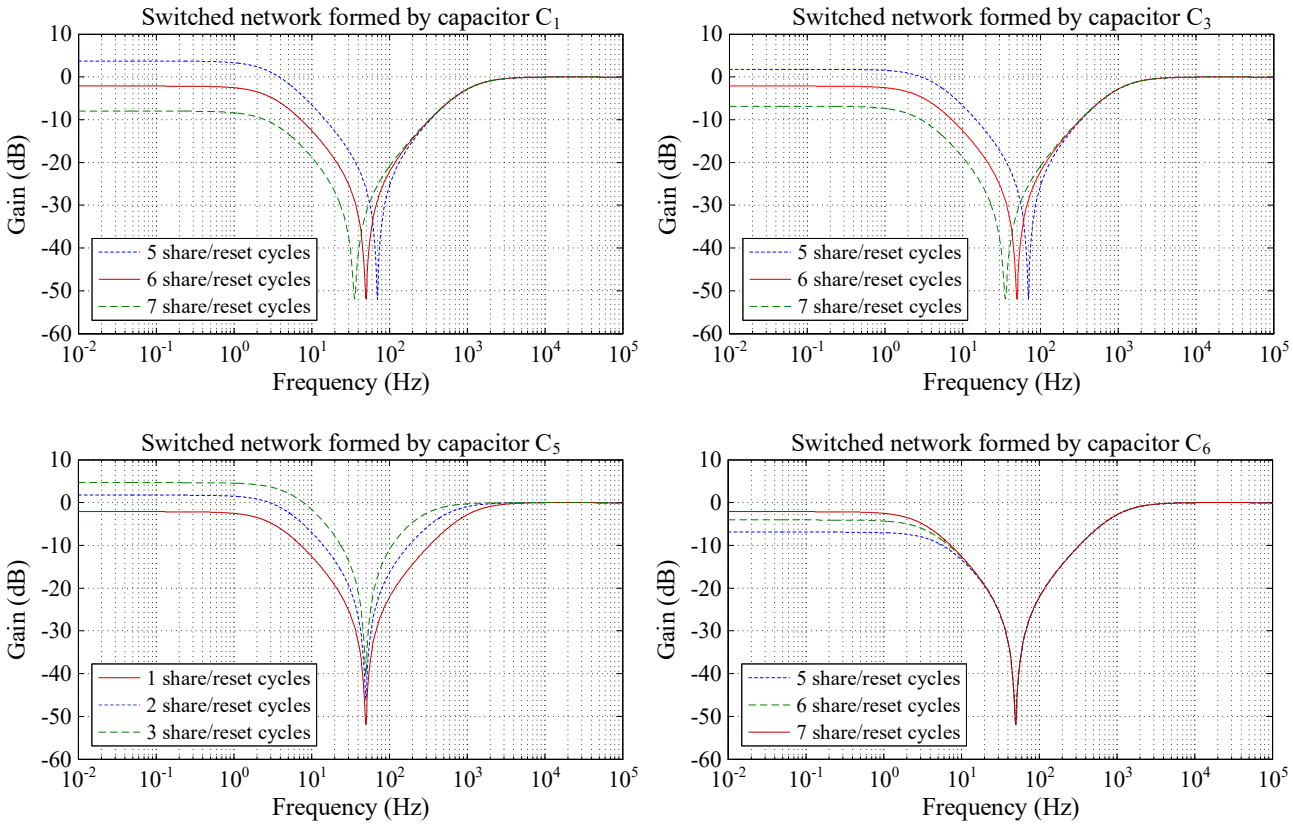


Figure 7.22: Post-layout frequency responses of the notch SC filter for different numbers of share/reset cycles.

A summary of the post-layout simulation results of some of the filter's metrics, for the different number of share/reset cycles, is shown in Table 7.8.

Table 7.8: Simulation summary of some of the filter metrics for different numbers of share/reset cycles.

Low Freq. Gain (dB)	Notch Frequency (kHz)	Notch Gain (dB)	Power (µW)
-8.2 ∼ 4.5	35.5 ∼ 70.0	-51.9 ∼ -40.2	272.6

The simulated output spectrum of the extracted (C+CC) notch SC filter is shown in Fig. 7.23, for the default number of share/reset cycles, where the input signal has two tones (f_1 = 50 Hz with 100 mV amplitude and f_2 = 5 kHz with 10 mV). Note that the output spectrum was normalized with the amplitude of the interference. The results shown that, even with the circuit's non-linear effects (ex. parasitic capacitances and switches ON resistance), the SC filter attenuates the 50 Hz interference in 52.3 dB, proving that this filter architecture achieves better results when compared with notch SC filters implemented with a bandpass filter and an adder [7], since this architecture requires very good matching between the gain of both blocks (at 50 Hz) and because it is very susceptible to the effects of the parasitic capacitances.

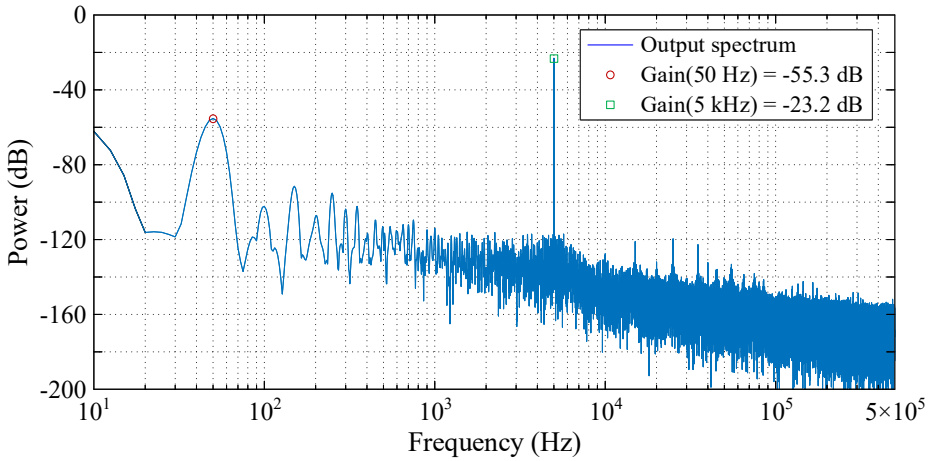


Figure 7.23: Output spectrum of the extracted notch SC filter with the default number of share/reset cycles for an input with two tones (f_1 = 50 Hz with 100 mV amplitude and f_2 = 5 kHz with 10 mV).

7.3 Conclusions

In this chapter, the prototypes of a programmable bandpass SC biquad filter using a low gain amplifier and of a 50 Hz notch SC filter using high gain amplifiers, implemented in UMC 130 nm CMOS technology, were described. The bandpass SC filter was optimized, using the non-hierarchical optimization methodology described in the previous chapter, to have a central frequency of approximately 4.5 kHz in the default bit configuration. Using control bits in some of the filter's switched networks, a certain degree of tunability in the filter's central frequency, gain, and quality factor was achieved. The frequency response obtained from the post-layout (C+CC) simulation and from the test board shows that the simulation software (Spectre) models the effect of the parasitic with a good degree of accuracy, allowing SC filters to be implemented using low gain amplifiers, instead of the typical SC filter architectures implemented with high-gain amplifiers.

The notch SC filter was designed, using the filter's ideal transfer function, to have the notch's frequency at 50 Hz. The large ratio between the sampling frequency and the notch frequency resulted in a very large dispersion between the capacitors' values. This large dispersion was reduced by a factor of 24 using the concept of charge division branches and up to 7 share/reset cycles, which allows larger capacitors to deliver a small amount of charge. This factor is well below the expected value (128) due to the effect of the switches parasitic capacitances, that also gets magnified by the number of share/reset cycles and needs to be compensated into the main capacitors, reducing their value. The results obtained from the simulation of the extracted (C+CC) circuit show that it is possible to achieve an attenuation of 52.3 dB.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

8.1 General Conclusions

The manual design of filter circuits is a time consuming process. The non-linear relation between the components dimensions and the multiple circuit specifications makes it a complex problem. Due to the large size of the design space and the competing specifications, it becomes difficult for designers to find a good compromise between all the circuit's specifications. Even after finding a good solution, it is necessary to test the sensitivity of the design to process, voltage supply, and temperature (PVT) variations and mismatch errors, to improve the circuit's yield after fabrication. This approach, however, substantially increases the number of simulations necessary to evaluate a possible solution, leading to the need of good analysis and optimization methodologies to better explore the design space and to minimize the design-to-fabrication time of a new circuit.

In Chapter 5, two methods for the design of switched-capacitor (SC) filters were presented. In the first method, the design is performed using a multi-step (hierarchical) approach. First the filter is designed from an ideal standpoint, using the filter's ideal transfer function. Afterwards, the amplifier is designed, using the inverse Laplace transform of its closed-loop transfer function for each clock phase. With this approach, it is possible to estimate the filter's capacitors load effect on the amplifier's output. Since this approach is based on continuous-time feedback theory, the effect of some of the circuit components are not correctly modeled, due to the switching nature of the circuit. For this reason, and to ensure that charge equilibrium is achieved in every clock phase, the amplifier's settling time should be slightly undersized. Finally, the switches are designed using a first-order RC model. Since this is a simple model, the time constants of the switches with a path to the amplifier's input or output should also be slightly undersized. In the case of parasitic-sensitive SC filters, after the design of these blocks, the parasitic capacitances from the amplifier and from the switches need to be compensated into ideal filter capacitor's design, to minimize the differences between the frequency response of the ideal design and of the filter at transistor level.

The second method uses a novel efficient numerical methodology to obtain a SC filter's frequency response, using the circuit's first-order differential equations. This methodology uses a non-hierarchical approach, where the non-ideal effects of the circuit's transistors are taken into consideration, using their medium frequency small-signal model, allowing the accurate computation of the filter's frequency response, even when charge equilibrium is not achieved by the end of a phase. The accuracy is demonstrated by comparing the results obtained using the proposed method with those obtained using a transient electrical simulation, in Spectre, for three SC circuits: a first-order passive SC filter, a second-order bandpass SC filter, and a second-order lowpass SC filter, showing that the results are in good agreement with the more time consuming electrical simulation of the circuits. Since the proposed method relies on simple matrix computations, it can be efficiently implemented in a computer, resulting in the quick computation of the filter's frequency response (< 1 s).

In Chapter 6, two optimization methodologies for SC filters, using genetic algorithms (GAs) and hybrid cost functions, were presented. In the hierarchical optimization methodology, the filter's frequency response is optimized using a multi-step approach. This results in the design space being broken into smaller, more manageable design spaces, making it easier to find a solution within the desired specifications. When optimizing SC filters implemented with low-gain amplifiers, due to the multi-step nature of this method and the sensitivity of the frequency response to the amplifier's gain, it becomes difficult to increase the robustness of the amplifier to PVT variations, since it is difficult to find a robust solution that has exactly the desired value. Three examples were given to demonstrate the performance of this optimization methodology: a second-order lowpass SC filter, used to demonstrate the benefits of using varying goal specifications to increase the convergence speed of the GA to the desired solution; a second-order bilinear bandpass SC filter, used to demonstrate the performance of the Monte Carlo (MC) optimization (mismatch errors); and a sixth-order bilinear bandpass SC filter, used to demonstrate an alternative approach to the optimization of high-order filters.

In the non-hierarchical optimization methodology, the filter's frequency response is optimized using a single system of equations, that accurately models the non-ideal behavior of circuit's transistors. Due to the size of the design space, which includes every circuit component, the GA will have a harder time finding good solutions at the beginning of the optimization process, when compared with the non-hierarchical method. For this reason, it is necessary to use varying goal specifications in the more demanding indicators, to increase the convergence speed of the GA. Since the amplifier's gain is not used as a design parameter, it is easier to increase the robustness of the frequency response to PVT variations, since the gain variations can be compensated using the filter's capacitors, maintaining the frequency response within the desired specifications. A lowpass and a bandpass SC filter biquads were used to demonstrate this optimization methodology's performance. Without PVT variations or mismatch errors, on average, the lowpass SC filter takes 33 generations (8 min) to find the final optimized solution and the bandpass SC filter takes 57 generations (28 min). A comparison between the hierarchical and non-hierarchical optimization methodologies, using the lowpass SC filter as a example, was also made in this chapter, showing a comparable performance in terms of optimization time.

In Chapter 7, the prototypes of a programmable second-order bandpass SC filter, using a low-gain amplifier, and of a 50 Hz notch SC filter, using high-gain amplifiers, were presented. These circuits were realized to be a part of an analog-to-digital (A/D) acquisition channel for an internet-of-things (IoT) water management sensor node. Both circuits were implemented with a supply voltage of 0.9 V in UMC 130 nm CMOS technology and were designed using the analysis and optimization methodologies described in Chapters 5 and 6.

Measurement results of the programmable bandpass SC filter show that the frequency response is in good agreement with the post-layout simulation. Using the five control bits, the filter's central frequency can be tuned between 3.9 kHz and 7.1 kHz, the gain between -6.4 dB and 12.6 dB, and quality factor between 0.9 and 6.9. The filter has a total power consumption of 249 μW and depending on the bit configuration, the circuit's THD is between -54.7 dB and -61.7 dB.

Using charge division branches in the implementation of the SC notch filter allowed an improvement in the capacitance spread by a factor of approximately 23. Transient simulations of the extracted view (C+CC) show that the amplitude of the 50 Hz interference was attenuated by 52.3 dB. The notch filter has a total power consumption of 273 μW.

8.2 Future Work

The work presented in this PhD thesis focused on analysis methodologies for the quick and accurate computation of a SC filter's frequency response. Optimization methodologies were also developed, using said analysis methods, for the optimization of the filter's frequency response and its sensitivity to PVT variations and mismatch errors. Special focus was given to implementations using low-gain amplifiers, since they can achieve higher bandwidths for the same power, when compared with implementations using high-gain amplifiers, and because filter architectures with more than one pole per amplifier can be used. The following research topics are suggested to further improve the analysis and optimization methodologies described in this thesis.

- The proposed optimization software mainly focuses on the optimization of the SC filter's frequency response, power consumption, and area. However, other performances metrics should be considered during the optimization process, such as the circuit's noise and distortion. Some research effort should be made to adapt the proposed non-hierarchical numerical analysis methodology to allow noise and distortion calculations.
- The optimization software optimizes SC filters based on the circuit's schematic netlist. However, this leaves the designer with the task of manually compensating the layout effects on the optimized schematic design. The software should be extended to include an estimation of the layout effects on the optimized design through the use of a back annotated netlist.
- Implementation of a co-optimization methodology, allowing the optimization, at system level, and the selection of the most appropriate topology for a given set of specifications before descending into the optimization at transistor level.

BIBLIOGRAPHY

[1] A. Baschirotto, V. Chironi, G. Cocciolo, S. D'Amico, M. De Matteis, and P. Delizia, "Low power analog design in scaled technologies", in *Proc. Topical Workshop Electron. Particle Phys. (TWEPP)*, Jan. 2009, pp. 103–109.

[2] Y. Zhao, P.-I. Mak, R. P. Martins, and F. Maloberti, "A 0.02 mm^2 59.2 dB SFDR 4th-order SC LPF with 0.5-to-10 MHz bandwidth scalability exploiting a recycling SC-buffer biquad", *IEEE J. Solid-State Circuits*, vol. 50, no. 9, pp. 1988–2001, Sep. 2015. DOI: 10.1109/JSSC.2015.2427334.

[3] H. Serra, R. Santos-Tavares, and N. Paulino, "A numerical methodology for the analysis of switched-capacitor filters taking into account non-ideal effects of switches and amplifiers", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 1, pp. 61–71, Jan. 2017. DOI: 10.1109/TCSI.2016.2601343.

[4] H. Serra, R. Santos-Tavares, and N. Paulino, "A top-down optimization methodology for SC filter circuit design", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2014, pp. 1672–1675. DOI: 10.1109/ISCAS.2014.6865474.

[5] H. Serra, R. Santos-Tavares, and N. Paulino, "A top-down optimization methodology for SC filter circuit design using varying goal specifications", in *Proc. Doctoral Conf. Computing, Elect., Ind. Syst. (DoCEIS)*, Apr. 2014, pp. 535–542. DOI: 10.1007/978-3-642-54734-8_59.

[6] H. Serra, R. Santos-Tavares, and J. Goes, "Automatic design of high-order SC filter circuits", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 1937–1940. DOI: 10.1109/ISCAS.2015.7169052.

[7] H. Serra, J. P. Oliveira, and N. Paulino, "A 50 Hz SC notch filter for IoT applications", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2017, pp. 2435–2438. DOI: 10.1109/ISCAS.2017.8050904.

[8] T. C. Carusone, D. A. Johns, and K. W. Martin, *Analog Integrated Circuit Design*, 2nd ed. John Wiley & Sons, 2012, ISBN: 9780470770108.

[9] J. T. Caves, M. A. Copeland, C. F. Rahim, and S. D. Rosenbaum, "Sampled analog filtering using switched capacitors as resistor equivalents", *IEEE J. Solid-State Circuits*, vol. 12, no. 6, pp. 592–599, Dec. 1977. DOI: 10.1109/JSSC.1977.1050966.

[10] G. Roubik and G. Temes, *Analog MOS integrated circuits for signal processing*. Wiley, 1986, ISBN: 9780471097976.

[11] H. Serra and N. Paulino, *Design of switched-capacitor filter circuits using low gain amplifiers*. Springer, 2015. DOI: 10.1007/978-3-319-11791-1.

[12] G. Wegmann, E. A. Vittoz, and F. Rahali, "Charge injection in analog MOS switches", *IEEE J. Solid-State Circuits*, vol. 22, no. 6, pp. 1091–1097, Dec. 1987. DOI: 10.1109/JSSC.1987.1052859.

[13] C. Eichenberger and W. Guggenbuhl, "Dummy transistor compensation of analog MOS switches", *IEEE J. Solid-State Circuits*, vol. 24, no. 4, pp. 1143–1146, Aug. 1989. DOI: 10.1109/4.34103.

[14] W. Yu and B. Leung, "Distortion and noise performance of bottom-plate sampling mixers", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 1998, pp. 401–404. DOI: 10.1109/ISCAS.1998.698883.

[15] K. W. Martin and A. S. Sedra, "Switched-capacitor building blocks for adaptive systems", *IEEE Trans. Circuits Syst.*, vol. 28, no. 6, pp. 576–584, Jun. 1981. DOI: 10.1109/TCS.1981.1085017.

[16] K. W. Martin and A. Sedra, "Effects of the op amp finite gain and bandwidth on the performance of switched-capacitor filters", *IEEE Trans. Circuits Syst.*, vol. 28, no. 8, pp. 822–829, Aug. 1981. DOI: 10.1109/TCS.1981.1085052.

[17] K. Nagaraj, T. Viswanathan, K. Singhal, and J. Vlach, "Switched-capacitor circuits with reduced sensitivity to amplifier gain", *IEEE Trans. Circuits Syst.*, vol. 34, no. 5, pp. 571–574, May 1987. DOI: 10.1109/TCS.1987.1086170.

[18] K. Haug, F. Maloberti, and G. C. Temes, "Switched-capacitor integrators with low finite-gain sensitivity", *Electron. Lett.*, vol. 21, no. 24, pp. 1156–1157, Nov. 1985. DOI: 10.1049/el:19850818.

[19] P. E. Allen and D. R. Holberg, "Switched capacitor circuits", in *CMOS Analog Circuit Design*, 2nd ed., Oxford University Press, 2002, ch. 9, pp. 492–611, ISBN: 9780195116441.

[20] B. J. Hosticka, R. W. Brodersen, and P. R. Gray, "MOS sampled data recursive filters using switched capacitor integrators", *IEEE J. Solid-State Circuits*, vol. 12, no. 6, pp. 600–608, Dec. 1977. DOI: 10.1109/JSSC.1977.1050967.

[21] K. W. Martin, "Improved circuits for the realization of switched-capacitor filters", *IEEE Trans. Circuits Syst.*, vol. 27, no. 4, pp. 237–244, Apr. 1980. DOI: 10.1109/TCS.1980.1084808.

[22] H. Schmid, "Offset, flicker noise, and ways to deal with them", in *Circuits at the Nanoscale: Communications, Imaging, and Sensing*, K. Iniewski, Ed., CRC Press, 2010, ch. 7, pp. 95–115, ISBN: 9781420070620.

[23] M. Ishikawa, R. Anzai, and N. Fujii, "Second order switched-capacitor filters reduced capacitance spread", in *Proc. of IEEJ Technical Meeting on Circuits and Syst.*, 1989, pp. 55–62.

[24] W. M. C. Sansen and P. Van Peteghem, "An area-efficient approach to the design of very-large time constants in switched-capacitor integrators", *IEEE J. Solid-State Circuits*, vol. 19, no. 5, pp. 772–780, Oct. 1984. DOI: 10.1109/JSSC.1984.1052220.

[25] N. Retdian, K. Takagi, and S. Takagi, "Capacitance spread reduction technique for low-cost switched-capacitor filters implementation", in *Proc. Eur. Conf. Circuit Theory and Design (ECCTD)*, Aug. 2009, pp. 37–40. DOI: 10.1109/ECCTD.2009.5275138.

[26] Q. Huang and W. Sansen, "Design techniques for improved capacitor area efficiency in switched-capacitor biquads", *IEEE Trans. Circuits Syst.*, vol. 34, no. 12, pp. 1590–1599, Dec. 1987. DOI: 10.1109/TCS.1987.1086078.

[27] K. Nagaraj, "A parasitic-insensitive area-efficient approach to realizing very large time constants in switched-capacitor circuits", *IEEE Trans. Circuits Syst.*, vol. 36, no. 9, pp. 1210–1216, Sep. 1989. DOI: 10.1109/31.34666.

[28] Y. Zhao, P.-I. Mak, M.-K. Law, and R. P. Martins, "A 0.127-mm^2, 5.6-mW, 5th-order SC LPF with +23.5-dBm IIP3 and 1.5-to-15-MHz clock-defined bandwidth in 65-nm CMOS", in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Nov. 2013, pp. 361–364. DOI: 10.1109/ASSCC.2013.6691057.

[29] Y. Zhao, P.-I. Mak, M.-K. Law, and R. P. Martins, "Improving the linearity and power efficiency of active switched-capacitor filters in a compact die area", *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 12, pp. 3104–3108, Dec. 2015. DOI: 10.1109/TVLSI.2014.2382674.

[30] A. Baschirotto, F. Montecchi, and R. Castello, "A 15 MHz 20 mW BiCMOS switched-capacitor biquad operating with 150 Ms/s sampling frequency", *IEEE J. Solid-State Circuits*, vol. 30, no. 12, pp. 1357–1366, Dec. 1995. DOI: 10.1109/4.482162.

[31] C.-Y. Wu, P.-H. Lu, and M.-K. Tsai, "Design techniques for high-frequency CMOS switched-capacitor filters using non-op-amp-based unity-gain amplifiers", *IEEE J. Solid-State Circuits*, vol. 26, no. 10, pp. 1460–1466, Oct. 1991. DOI: 10.1109/4.90103.

[32] H. H. Boo, D. S. Boning, and H.-S. Lee, "A 12b 250 MS/s pipelined ADC with virtual ground reference buffers", *IEEE J. Solid-State Circuits*, vol. 50, no. 12, pp. 2912–2921, Dec. 2015. DOI: 10.1109/JSSC.2015.2467183.

[33] H. H. Boo, *Virtual Ground Reference Buffer Technique in Switched-Capacitor Circuits*. Massachusetts Institute of Technology, Jun. 2015. [Online]. Available: http://hdl.handle.net/1721.1/99812.

[34] A. de la Plaza, "High-frequency switched-capacitor using unity-gain buffers", *IEEE J. Solid-State Circuits*, vol. 21, no. 3, pp. 470–477, Jun. 1986. DOI: 10.1109/JSSC.1986.1052553.

[35] R. P. Sallen and E. L. Key, "A practical method of designing RC active filters", *IRE Trans. Circuit Theory*, vol. 2, no. 1, pp. 74–85, Mar. 1955. DOI: 10.1109/TCT.1955.6500159.

[36] W. R. Bennett, "Steady-state transmission through networks containing periodically operated switches", *IRE Trans. Circuit Theory*, vol. 2, no. 1, pp. 17–21, Mar. 1955. DOI: 10.1109/TCT.1955.6500148.

[37] M. L. Liou, "Exact analysis of linear circuits containing periodically operated switches with applications", *IEEE Trans. Circuit Theory*, vol. 19, no. 2, pp. 146–154, Mar. 1972. DOI: 10.1109/TCT.1972.1083438.

[38] T. Ström and S. Signell, "Analysis of periodically switched linear circuits", *IEEE Trans. Circuits Syst.*, vol. 24, no. 10, pp. 531–541, Oct. 1977. DOI: 10.1109/TCS.1977.1084274.

[39] M. L. Liou and Y.-L. Kuo, "Exact analysis of switched capacitor circuits with arbitrary inputs", *IEEE Trans. Circuits Syst.*, vol. 26, no. 4, pp. 213–223, Apr. 1979. DOI: 10.1109/TCS.1979.1084633.

[40] R. Frasca, M. Kanat Camlibel, I. Cem Goknar, L. Iannelli, and F. Vasca, "Linear passive networks with ideal switches: Consistent initial conditions and state discontinuities", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 12, pp. 3138–3151, Dec. 2010. DOI: 10.1109/TCSI.2010.2052511.

[41] J. Ruiz-Amaya, M. Delgado-Restituto, and A. Rodríguez-Vázquez, "Accurate settling-time modeling and design procedures for two-stage Miller-compensated amplifiers for switched-capacitor circuits", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 6, pp. 1077–1087, Jun. 2009. DOI: 10.1109/TCSI.2008.2008509.

[42] P. Malcovati, S. Brigati, F. Francesconi, F. Maloberti, P. Cusinato, and A. Baschirotto, "Behavioral modeling of switched-capacitor sigma-delta modulators", *IEEE Trans. Circuits Syst. I, Fundam. Theory and Appl.*, vol. 50, no. 3, pp. 352–364, Mar. 2003. DOI: 10.1109/TCSI.2003.808892.

[43] J. Ruiz-Amaya, J. M. de la Rosa, F. V. Fernández, F. Medeiro, R. del Río, B. Pérez-Verdú, and A. Rodríguez-Vázquez, "High-level synthesis of switched-capacitor, switched-current and continuous-time ΣΔ modulators using SIMULINK-based time-domain behavioral models", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 9, pp. 1795–1810, Sep. 2005. DOI: 10.1109/TCSI.2005.852479.

[44] Y. Tsividis, "Principles of operation and analysis of switched-capacitor circuits", *Proc. IEEE*, vol. 71, no. 8, pp. 926–940, Aug. 1983. DOI: 10.1109/PROC.1983.12699.

[45] E. Hökenek and G. Moschytz, "Analysis of general switched-capacitor networks using indefinite admittance matrix", *IEE Proc. G, Electron. Circuits and Syst.*, vol. 127, no. 1, pp. 21–33, Feb. 1980. DOI: 10.1049/ip-g-1.1980.0004.

[46] C. F. Kurth and G. S. Moschytz, "Two-port analysis of switched-capacitor networks using four-port equivalent circuits in the z-domain", *IEEE Trans. Circuits Syst.*, vol. 26, no. 3, pp. 166–180, Mar. 1979. DOI: 10.1109/TCS.1979.1084627.

[47] C. F. Kurth and G. S. Moschytz, "Nodal analysis of switched-capacitor networks", *IEEE Trans. Circuits Syst.*, vol. 26, no. 2, pp. 93–105, Feb. 1979. DOI: 10.1109/TCS.1979.1084613.

[48] C. W. Ho, A. E. Ruehli, and P. A. Brennan, "The modified nodal approach to network analysis", *IEEE Trans. Circuits Syst.*, vol. 22, no. 6, pp. 504–509, Jun. 1975. DOI: 10.1109/TCS.1975.1084079.

[49] G. D. Hachtel, R. K. Brayton, and F. G. Gustavson, "The sparse tableau approach to network analysis and design", *IEEE Trans. Circuit Theory*, vol. 18, no. 1, pp. 101–113, Jan. 1971. DOI: 10.1109/TCT.1971.1083223.

[50] J. Vandewalle, H. J. de Man, and J. Rabaey, "Time, frequency, and z-domain modified nodal analysis of switched-capacitor networks", *IEEE Trans. Circuits Syst.*, vol. 28, no. 3, pp. 186–195, Mar. 1981. DOI: 10.1109/TCS.1981.1084971.

[51] A. Konczykowska and M. Bon, "Automated design software for switched-capacitor IC's with symbolic simulator SCYMBAL", in *Proc. 25th ACM/IEEE Design Autom. Conf.*, 1988, pp. 363–368, ISBN: 0818688645.

[52] M. Helena Fino, J. E. Franca, and A. Steiger-Garção, "Automatic symbolic analysis of switched-capacitor filtering networks using signal flow graphs", *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 14, no. 7, pp. 858–867, Jul. 1995. DOI: 10.1109/43.391733.

[53] F. Yuan and A. Opal, "Computer methods for switched circuits", *IEEE Trans. Circuits Syst. I, Fundam. Theory and Appl.*, vol. 50, no. 8, pp. 1013–1024, Aug. 2003. DOI: 10.1109/TCSI.2003.815193.

[54] R. Plodeck, U. W. Brugger, D. C. Von Grünigen, and G. S. Moschytz, "SCANAL - A program for the computer-aided analysis of switched-capacitor networks", *IEE Proc. G, Electron. Circuits and Syst.*, vol. 128, no. 6, pp. 277–285, Dec. 1981. DOI: 10.1049/ip-g-1.1981.0063.

[55] H. de Man, J. Rabaey, L. Claesen, and J. Vandewalle, "DIANA-SC : A complete CAD system for switched capacitor filters", in *Proc. European Solid-State Circuits Conf. (ESSCIRC)*, Sep. 1981, pp. 130–133.

[56] H. J. de Man, J. Rabaey, G. Arnout, and J. Vandewalle, "Pratical implementation of a general computer aided design technique for switched capacitor circuits", *IEEE J. Solid-State Circuits*, vol. 15, no. 2, pp. 190–200, Apr. 1980. DOI: 10.1109/JSSC.1980.1051362.

[57] J. Vandewalle, H. J. de Man, and J. Rabaey, "The adjoint switched capacitor network and its application to frequency, noise and sensitivity analysis", *Int. J. Circuit Theory Appl.*, vol. 9, no. 1, pp. 77–88, Jan. 1981. DOI: 10.1002/cta.4490090109.

[58] S. C. Fang, Y. P. Tsividis, and O. Wing, "SWITCAP: A switched-capacitor network analysis program part I: Basic features", *IEEE Circuits Syst. Mag.*, vol. 5, no. 3, pp. 4–10, Sep. 1983. DOI: 10.1109/MCAS.1983.6323864.

[59] S. C. Fang, Y. P. Tsividis, and O. Wing, "SWITCAP: A switched-capacitor network analysis program part II: Advanced applications", *IEEE Circuits Syst. Mag.*, vol. 5, no. 4, pp. 41–46, Dec. 1983. DOI: 10.1109/MCAS.1983.6323886.

[60] Y. Tsividis, "Analysis of switched capacitive networks", *IEEE Trans. Circuits Syst.*, vol. 26, no. 11, pp. 935–947, Nov. 1979. DOI: 10.1109/TCS.1979.1084588.

[61] S. C. Fang and Y. Tsividis, "Modified nodal analysis with improved numerical methods for switched capacitive networks", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Apr. 1980, pp. 977–980.

[62] S. C. Fang, Y. Tsividis, and O. Wing, "Analysis of SCN's with nonlinear and time-varying elements using time-invariant charge variables", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 1982, pp. 1–4.

[63] C. F. Lee, R. D. Davis, W. K. Jenkins, and T. N. Trick, "Sensitivity and nonlinear distortion analyses for switched- capacitor circuits using SCAPN", *IEEE Trans. Circuits Syst.*, vol. 31, no. 2, pp. 213–221, Feb. 1984. DOI: 10.1109/TCS.1984.1085476.

[64] R. D. Davis, T. N. Trick, and W. K. Jenkin, "An efficient LU factorization scheme for the analysis of switched-capacitor filters", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 1982, pp. 33–37.

[65] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. Krieger Publishing Company, 1980, ISBN: 9781575242835.

[66] L. Yue and S. Y. Chao, "SCNSOP: A switched-capacitor circuit simulation and optimisation program", *IEE Proc. G, Electron. Circuits and Syst.*, vol. 133, no. 2, pp. 107–112, Apr. 1986. DOI: 10.1049/ip-g-1.1986.0015.

[67] A. Opal and J. Vlach, "Analysis and sensitivity of periodically switched linear networks", *IEEE Trans. Circuits Syst.*, vol. 36, no. 4, pp. 522–532, Apr. 1989. DOI: 10.1109/31.92884.

[68] J. Vlach, K. Singhal, and M. Vlach, "Computer oriented formulation of equations and analysis of switched-capacitor networks", *IEEE Trans. Circuits Syst.*, vol. 31, no. 9, pp. 753–765, Sep. 1984. DOI: 10.1109/TCS.1984.1085583.

[69] K. Kundert. (Jun. 2015). Simulating switched-capacitor filters with SpectreRF, [Online]. Available: http://www.designers-guide.com/Analysis//sc-filters.pdf.

[70] W.-H. Ki and G. C. Temes, "Optimal capacitance assignment of switched-capacitor biquads", *IEEE Trans. Circuits Syst. I: Fundam. Theory and Appl.*, vol. 42, no. 6, pp. 334–342, Jun. 1995. DOI: 10.1109/81.390265.

[71] H. Tang, "A systematic design flow for optimal capacitance assignment in switched-capacitor biquads", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 7, pp. 2076–2086, Feb. 2008. DOI: 10.1109/TCSI.2008.918264.

[72] G. Shi, A. Zhang, and Y. Gu, "A comparative study on using linear programming and simulated annealing in the optimal realization of a SC filter", *Int. J. Circuit Theory Appl.*, 2017. DOI: 10.1002/cta.2350.

[73] P. E. Fleischer and K. R. Laker, "A family of active switched capacitor biquad building blocks", *Bell Syst. Tech. J.*, vol. 58, no. 10, pp. 2235–2269, Dec. 1979. DOI: 10.1002/j.1538-7305.1979.tb02965.x.

[74] R. Santos-Tavares, N. Paulino, J. Goes, and J. P. Oliveira, "Optimum sizing and compensation of two-stage CMOS amplifiers based on a time-domain approach", in *Proc. IEEE Int. Conf. on Electronics, Circuits Systems (ICECS)*, Dec. 2006, pp. 533–536. DOI: 10.1109/ICECS.2006.379843.

[75] A. S. Sedra and K. C. Smith, "Feedback", in *Microelectronic Circuits*, 6th ed., Oxford University Press, 2009, ch. 8, pp. 802–909, ISBN: 9780195323030.

[76] R. A. Adams and C. Essex, "Techniques of integration", in *Calculus: A Complete Course*, 7th ed., Pearson Education Canada, 2010, ch. 6, pp. 340–345, ISBN: 9780321549280.

[77] H. Serra, N. Paulino, and J. Goes, "A switched-capacitor biquad using a simple quasi-unity gain amplifier", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2013, pp. 1841–1844. DOI: 10.1109/ISCAS.2013.6572223.

[78] J. Lebl, "Systems of ODEs", in *Notes on Diffy Qs: Differential Equations for Engineers*, CreateSpace Independent Publishing Platform, 2014, ch. 3, pp. 124–128, ISBN: 9781541329058.

[79] D. S. Watkins, "Eigenvalues and eigenvectors II", in *Fundamentals of Matrix Computations*, 2nd ed., John Wiley and Sons, 2002, ch. 6, pp. 502–519, ISBN: 9780471461678.

[80] H. Serra, N. Paulino, and J. Goes, "A switched-capacitor band-pass biquad filter using a simple quasi-unity gain amplifier", in *Proc. Doctoral Conf. Computing, Elect., Ind. Syst. (DoCEIS)*, Apr. 2013, pp. 582–589. DOI: 10.1007/978-3-642-37291-9_63.

[81] M. Figueiredo, R. Santos-Tavares, E. Santin, J. Ferreira, G. Evans, and J. Goes, "A two-stage fully differential inverter-based self-biased CMOS amplifier with high efficiency", *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 7, pp. 1591–1603, Jul. 2011. DOI: 10.1109/TCSI.2011.2150910.

[82] R. Santos-Tavares, N. Paulino, and J. Goes, *Time-domain optimization of CMOS amplifiers: Based on distributed genetic algorithms*. LAP LAMBERT Academic Publishing, 2012, ISBN: 9783847329251.

[83] P. Nenzi and H. Vogt. (Jan. 2014). Ngspice users manual version 26, [Online]. Available: http://ngspice.sourceforge.net/.

[84] N. Paulino, J. Goes, and A. Steiger-Garção, "Design methodology for optimization of analog building blocks using genetic algorithms", in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2001, pp. 435–438. DOI: 10.1109/ISCAS.2001.922078.

[85] M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, P. Alken, M. Booth, F. Rossi, and R. Ulerich. (Jun. 2017). GNU scientific library, [Online]. Available: http://www.gnu.org/software/gsl/.

[86] (2015). Proteus - adaptative microfluidic and nano-enabled smart systems for water quality sensing, [Online]. Available: http://www.proteus-sensor.eu/.

[87] K. Gopalan and T. Bourouina, "Water conductivity sensor characterization", ESIEE Paris/ESYCOM, Tech. Rep., 2016.

[88] R. J. Baker, *CMOS circuit design, layout, and simulation*, 3rd ed. Wiley-IEEE Press, 2010, ISBN: 9780470881323.

[89] O. Choksi and L. R. Carley, "Analysis of switched-capacitor common-mode feedback circuit", *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 50, no. 12, pp. 906–917, Dec. 2003. DOI: 10.1109/TCSII.2003.820253.