NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

DUARTE GONÇALVES LOPES DA SILVA

Master/BSc in Electrical and Computer Engineering

SOLVING DIFFERENTIAL EQUATIONS IN ANALOG DOMAIN

MASTER IN MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

NOVA University Lisbon
September, 2023

NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

DEPARTMENT OF ELECTRICAL
AND COMPUTER ENGINEERING

SOLVING DIFFERENTIAL EQUATIONS IN ANALOG DOMAIN

DUARTE GONÇALVES LOPES DA SILVA

Master/BSc in Electrical and Computer Engineering

Adviser: Luís Bica Oliveira
Associate Professor, NOVA University Lisbon

Examination Committee

Chair: Anikó Katalin Horváth da Costa
Full Professor, FCT-NOVA

Rapporteur: Bruno João Nogueira Guerreiro
Associate Professor, Another University

MASTER IN MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA
E DE COMPUTADORES

NOVA University Lisbon
September, 2023

Solving differential equations in analog domain

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v6.10.10) [1].

Acknowledgements

A realização deste trabalho não teria sido possivel sem o apoio e a coperação de diversas pessoas, às quais gostaria de fazer um especial agradecimento.

Quero começar por agradecer às duas pessoas que me orientaram no desenvolvimento desta tese. Primeiro, o meu orientador, o professor Luís Oliveira por toda a sua disponibilidade e apoio. Segundo, o meu colega João Araújo, por toda a sua ajuda que foi além do que alguma vez seria de esperar.

A todos os professores do DEEC com os quais me cruzei ao longo destes cinco anos de formação.

A todos os meus amigos que me acompanharam ao longo deste percurso.

Por último, mas não menos importante, para a minha familia que me apoia sempre a partir dos bastidores. Em especial, para a minha namorada, que muito me aturou e incentivou para completar esta etapa.

"No great discovery was ever made without a bold guess." (Newton)

Abstract

A growth in the use of analog computers has been seen in the last few years despite of the dominance of digital computers, this is due to the fact that it was noticed that analog computers could solve differential equations not only faster but also in a more efficient way, which means drawing less power. This lead to the interest in studying how a computer can solve this type of equations and what it takes to do so.

With this in mind a circuit that describes a variation of the law of exponential decay is proposed, this implicated the design of an folded cascode operational amplifier, which was implemented using 130nm MOS technology. This then led to the study of this amplifier in an integrator configuration as well as the proposed circuit, which in a later stage was made configurable by adding a logic controller to control the resistors that establish the coefficients of the equation.

With the obtained results it was possible to validate the objective specifications for the amplifier as well as observe that the circuit was able to come to successful results within the known constraints of the components used for this circuit. This circuit was able to reach solutions much faster than a digital computer facing the same problem, 30μs versus 0.3s, while consuming considerably less power, 40mW versus 72W.

Keywords: Analog computing, Differential equation solver, CMOS, Folded-Cascode, Integrator

Resumo

Nos últimos anos tem se verificado um aumento no uso de computadores analógicos apesar do contínuo domínio por parte dos computadores digitais, isto deve-se principalmente ao facto de que foi observado que os computadores analógicos conseguem não só resolver equações diferenciais muito mais rápidos, mas também que estes conseguem fazê-lo de uma maneira muito mais eficiente, ou seja, gastando menos potência. Isto levou ao interesse em estudar como estes computadores resolvem estas equações e também o que é necessário para o fazer.

Devido as estes fatores, um circuito que representa uma variante da lei de decaimento exponencial foi proposto para ser resolvido, isto implicou o dimensionamento de um amplificador operacional de topologia "folded-cascode", que foi implementado usando tecnologia MOS de 130nm. Isto levou ao estudo deste amplificador na sua montagem integradora para além do circuito proposto, por fim este circuito foi tornado configurável por meio de um controlador lógico adicional a controlar as resistências que determinam os valores dos coeficientes desta equação.

Com os resultados obtidos, foi não só possível validar que as características definidas para o amplificador foram cumpridas como também observar que o circuito conseguiu obter resultados bem sucedidos dentro das limitações dos componentes do circuito. Assim este circuito conseguiu chegar ao resultado destas equações num tempo muito inferior ao de um computador digital, 30μs contra 0.3s, enquanto consumia uma potência estimada consideravelmente mais baixa, 40mW contra 72W.

Palavras-chave: Computação analógica, Equação diferencial, Amplificador, Integrador

Contents

List of Figures ix

List of Tables xi

Acronyms xiii

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Main contributions and objective . . . 2
- 1.3 Thesis organization . . . 2

2 Background 3
- 2.1 Analog computing . . . 3
- 2.2 Differential Equations . . . 5
- 2.3 Solver with current variable . . . 7
  - 2.3.1 Integrator Blocks . . . 7
  - 2.3.2 Fan-out Blocks . . . 8
  - 2.3.3 Variable Gain Amplifiers/Multiplier Blocks . . . 8
- 2.4 Solver with voltage variable . . . 9
  - 2.4.1 Methods . . . 10
  - 2.4.2 Internal and Boundary Modules . . . 11

3 State-of-Art 12
- 3.1 Current-based solver . . . 12
- 3.2 Voltage-based solver . . . 15
  - 3.2.1 One dimension wave equation solver . . . 16
  - 3.2.2 Transmission line problem solver . . . 18
  - 3.2.3 Variable area duct problem solver . . . 20
- 3.3 Solver results . . . 22
- 3.4 Solver comparison and discussion . . . 23

4 Proposed circuit and analysis **25**
4.1 Amplifier . . . 26
4.1.1 Required characteristics . . . 28
4.1.2 Relevant equations . . . 28
4.1.3 Mathcad and Final design . . . 30
4.2 Integrator analysis . . . 33
4.3 Proposed circuit . . . 35
4.3.1 Linear differential equation . . . 35
4.3.2 Law of exponential decay . . . 37
4.3.3 Making it configurable . . . 38

5 Simulation results and validation **41**
5.1 Operational amplifier . . . 42
5.1.1 DC gain . . . 43
5.1.2 Gain Bandwidth Product . . . 44
5.2 Integrator simulations . . . 44
5.2.1 Sine wave integration . . . 45
5.2.2 Square wave integration . . . 48
5.3 Linear differential equation . . . 51
5.3.1 No configuration . . . 51
5.3.2 With configuration . . . 54

6 Conclusions and Future work **59**
6.1 Conclusions . . . 59
6.2 Future work . . . 60

Bibliography **61**

List of Figures

2.1 Comparison between an old and new analog computer 4
2.2 Two differential equations examples 6
2.3 Integrator block 7
2.4 Comparison between two multipliers 9

3.1 Current-based architecture of an analog computer, adapted from [3] 13
3.2 Integrator schematic 14
3.3 Functionalities of the digital side of this design, extracted from [3] 15
3.4 Example of a coupled mass-spring problem, adapted from [3] 15
3.5 Design of an architecture implemented with IM and BM 16
3.6 Schematic of the used internal module, adapted from [26] 18
3.7 Model of a transmission line problem, adapted from [29] 18
3.8 Design of the internal module, extracted from [29] 20
3.9 Design of the internal module, adapted from [25] 21
3.10 Variable area duct problem, extracted from [25] 22

4.1 Amplifier circuit and it's biasing circuit 27
4.2 Amplifier gain as a function of vdsat and transistor length 30
4.3 Gain Bandwidth Product as a function of vdsat and transistor length 31
4.4 Noise as a function of vdsat 31
4.5 Frequency of the second pole as a function of vdsat and transistor length . . 32
4.6 Opamp in an integrator montage 33
4.7 Opamp in an integrator montage with a feedback resistor 34
4.8 Circuit of a linear differential equation, extracted from [35] 36
4.9 Design used for the switchable resistors 38
4.10 Circuit designed to control the switches 39
4.11 Parameters that allow for the proper function of the switch 39

5.1 Schematic of a folded cascode in Cadence's software 42
5.2 Schematic of the test-bench used for simualtions 42

5.3 DC gain of the folded cascode amplifier 43
5.4 GBW of the folded cascode amplifier 44
5.5 Frequency response of the integrator montage 45
5.6 Parameters and schematic used for the simulation of the sine wave 47
5.7 Result of the sine wave integration 48
5.8 Parameters used for the simulation of the square wave and it's result 49
5.9 Simulation results from the integration of a square wave 50
5.10 Circuit that defines the differential equation 52
5.11 Comparison between the solutions 53
5.12 Simulation results in the first condition 56
5.13 Simulation results in the second condition 57
5.14 Simulation results in the last condition 58

List of Tables

3.1 Accuracy results of the current-based analog computer 22
3.2 Accuracy results of each voltage-based analog computer 23
3.3 Difference in speed of the analog to digital computer 23

4.1 Circuit results after the tuning of the transistor parameters 32
4.2 Final sizing of the circuit transistors 32

5.1 Comparison of solution time and power consumption 54
5.2 Resistor values for test 1 55
5.3 Resistor values for test 2 55
5.4 Resistor values for the last test 55
5.5 Expressions for test 1 55
5.6 Expressions for test 2 55
5.7 Expressions of the last test 55
5.8 Simulations results compared to ideal results for the first test 56
5.9 Simulations results compared to ideal results for the second test 57
5.10 Simulations results compared to ideal results for the third test 58

Acronyms

ADC Analog to Digital Converter *(pp. 14, 20, 39)*

CMOS Complementary Metal-Oxide Semiconductor *(pp. 1–3, 5, 8, 9, 17, 21, 59)*
CPU Central Processing Unit *(pp. 12, 23)*

DAC Digital to Analog Converter *(pp. 13, 14, 20)*
DC Direct Current *(pp. 11, 13, 16, 17, 19, 28, 29, 36, 39, 43, 51, 52)*

FCA Folded Cascode Amplifier *(pp. 26, 28, 43, 59, 60)*
FPGA Field Programmable Gate Array *(pp. 12, 16)*

GBW Gain Bandwidth Product *(pp. 9, 17, 28, 29, 31, 44, 45)*
GPU Graphics Processing Unit *(pp. 12, 23)*

MSD Mean Square Difference *(p. 22)*

opamp Operational Amplifier *(pp. 2, 5, 9, 11, 17, 21, 22, 25–27, 29, 30, 32, 33, 36, 41, 43, 44, 46, 54, 59)*
OTA Operational Transconductance Amplifier *(pp. 8, 13)*

RMS Root Mean Squared *(p. 22)*

SoC System on a Chip *(p. 20)*
SPSA Simultaneous Perturbation Stochastic Approximation *(p. 18)*

VGA Variable Gain Amplifier *(pp. 5, 8)*

1 Introduction

1.1 Motivation

Analog computers started being used way before digital technology emerged, they go as far back as the mid-20th century [2, 3], but they gained especial importance during World War II where they were vastly used to control and operate much of the weaponry, such as the targeting systems of the V-2 rocket or the M9 gun director [4]. As the time went on these computers got bigger in size and harder to program which made digital computers take over for problem solving.

Following the last five decades digital computers have since surpassed analog computers and have been continuously getting better following Moore's Law. For most applications, especially general purpose and large scale ones, they have completely taken over due to their flexibility in problem solving. In the attempt of solving these problems with digital computers many methods have been used. Discrete-domain numerical methods, like finite difference [5–7] and method of the moments [8], that work by discretizing the spacial domain into spacial grids. But also those that discretize in the time domain like the Runge-Kutta method [9, 10].

Although these methods make it so that the solutions are precise and exact, they also slow down the computer, making it so that it takes many clock cycles even in the more advanced and high-speed digital processors [11].

Over the last few years, the important characteristics of a digital computer like clock speeds and power usage have been pushed to their limits, making it look like Moore's Law is beginning to look less relevant in today's scenario [12]. Even the transistor scaling has come to a point where new technologies have to be developed, like FinFets [13]and GAAFets, this slowing down of the digital technologies is still applicable, although it will be interesting to revisit analog computers with these new nano-Complementary Metal-Oxide Semiconductor (CMOS) designs. With this came the resurfacing of analog computers which will be the focus of this dissertation thesis but many other options, like optical computing [14, 15], which solves thermal and bandwidth constraints, but also memristor based computing which looks promising in machine learning operations and

for solving partial differential equations [16].

Proved by the prototype from Colombia University [2, 17–19], as an example of a recent hybrid computer, analog technologies have been brought back to solve targeted problems in a low-cost and energy-efficient way realized in widely available CMOS technologies at the cost of a lower precision in the end result of the problem. Some are even studying the possibility of a hybrid computer where firstly an initial solution is computed by an analog computer so that then the digital computer can apply its iterative process using that initial solution to get a more precise result, thus speeding up the process compared to it solving the problem from the start [17].

1.2 Main contributions and objective

In this dissertation there is the objective to study how an analog computer can used to solve differential equations which represent most of the problems of today. With this in mind an Operational Amplifier (opamp), which is the base of any computer, is going to be designed, so that on a later stage it can be studied in it's integrator configuration.

A circuit that represents a differential equation is going to be analyzed and presented so that simulations can be done to validate it's effectiveness in solving this equation, lastly this circuit will be made configurable so that it can solve multiple variations of this equation.

1.3 Thesis organization

This thesis is organized in 6 chapters, including the present which gives a brief introduction and motivation to the theme of this dissertation. Then Chapter 2 will show different approaches to analog computing and the problems they solve. From Chapter 3 you can expect to find the more relevant analog computers found and a more in-depth comparison between them.

On chapter 4 the proposed circuit and problem are going to be presented as well as the components being used will be designed. Moving to chapter 5 the simulations done to validate the correct function of these components will be shown as well as the ones that show that the circuit can solve the proposed problem. Finally the last chapter 6 will reveal the conclusions taken from realizing this work as well as what type of future work it can lead to.

2

Background

In this section of the thesis some background is going to be given to the key topics relevant to this work, starting off with the most important that is analog computing and how has it evolved over the years, then some information about how differential equations are used to solve must of the problems that are encountered in our daily lives. Then it shifts more in depth into the two types of current analog computers and what components and techniques they rely on.

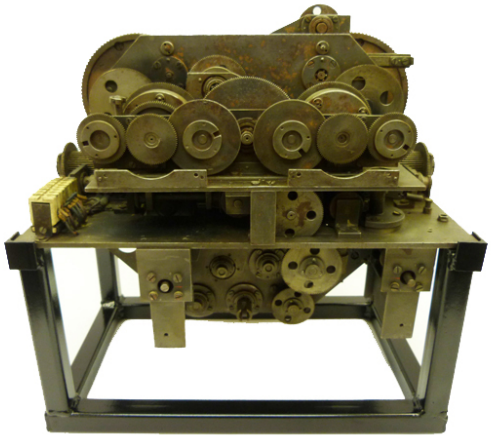
2.1 Analog computing

When analog computers started being used they were the only way to solve the problems of the time, now they are looked at as a less expensive way, due to the use of more mainstream CMOS technologies, to solve more focused problems.

These computers work by instead of encoding the problem's numbers using binary code, they use the circuit variables, voltage and current, to encode them. They also operate continuously through time compared to digital computers which work in steps determined by the clock of the hardware.

With most of science's problems, in areas like fluid dynamics, thermodynamics and electromagnetics, being mostly represented by a set of partial differential equations or ordinary differential equations, as long as they can be put into a tensor equation this system can be modeled by an analog computer.

Analog computers started off by using mechanical and vacuum solutions, but to program these required physical input, like turning knobs or toggling switches [4]. Nowadays to model the previously mentioned problems electronic analog computers are used and these can be either passive or active depending on the components they use. Figure 2.1 shows the difference between an analog computer used as a guidance system for a rocket by the germans in the second world war compared to a modern analog computer that can be used to model a variety of different problems.



(a) V-2 Rocket guidance system.

(b) Modular analog computer.

Figure 2.1: Comparison between an old and new analog computer.

In a passive analog computer it is possible to capture the spatial characteristic of the modeled physical quantities, for instance voltage, power or temperature, by using a network of both resistors and reactive elements it is possible to design a suitable circuit that describes the problem accordingly. Due to the fact that making both high quality inductors and transformers in integrated circuit is difficult, which would be a constraint for the circuits maximum operating frequency, this type of analog computer has been surpassed by the active type, even though in continuous-time computation it is possible to achieve a more accurate result by implementing the circuit with passive elements due to the fact that these are overall less noisy.

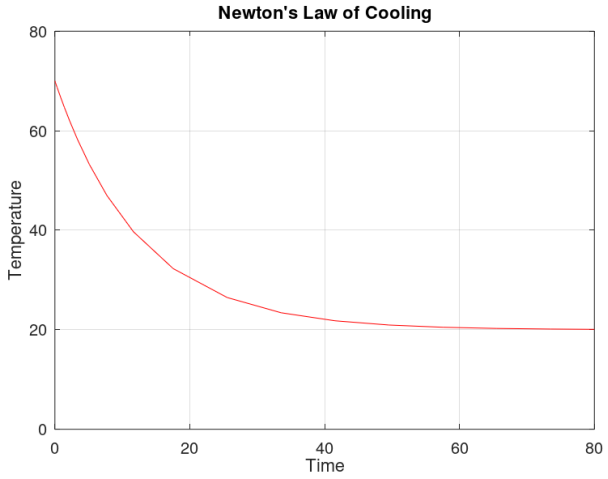
In turn, with an active analog computer that is generally made using opamps which enables not only accurate solutions to the problems but it also allows for a more flexible device, it can be reconfigured to solve other types of problems with ease. By letting go of the passive elements it is possible to get a circuit that is easier to fabricate in CMOS integrated circuit technologies and also in certain designs that will be discussed in Chapter 3, problems such as losses and delays that occur within the circuit elements will be made up due to the design of the chosen circuit to implement the problem.

Two different approaches have been pursued in the modeling of a circuit to solve differential equations, current base, consists of a grid of integrators, Variable Gain Amplifier (VGA)s/multipliers and fan-out blocks that by summing the currents in each node of grid create the coefficients of the solution to the problem. Meanwhile a voltage based solver which consists of a series of internal modules connected to each other and surrounded by boundary modules that are preloaded with boundary conditions for each problem.

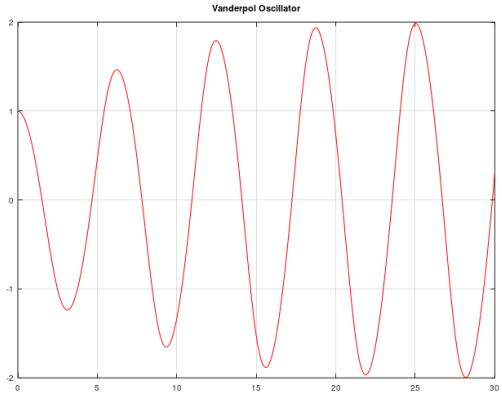
2.2 Differential Equations

Differential equations have been at the core of most of the relevant problems being solved nowadays, these having numerous applications across various fields, such as engineering, physics, biology and computer science. When trying to understand and predict the behaviour of a dynamic system these equations provide a powerful tool by describing how the variable changes in relation to itself, other variables or time [20].

These equations can take many shapes from the most simple first order linear equation, to the more complex higher order non-linear equations. They can also be ordinary or partial depending on if they have one or more independent variables. The first order linear equation could take the form of the Newton's Law of Cooling, which in itself is a specific case of the law of exponential decay, where this simple equation describes the rate of cooling of a body until it reaches the environment's temperature around it. Or the higher order non-linear equation, which could represent an oscillator following the Van der Pol's equation, where there is a second order term which introduces some non-linearities.



(a) Newton's law of cooling result example.



(b) Example of a Van der Pol oscillator.

Figure 2.2: Comparison between the simpler first order linear equation and a more complex second order non-linear equation

With this in mind and the advancements that have been done in computer technology, such has the exponential increase in computational performance, the creation of numerical methods and the more accessible specific hardware, not only did it become possible solve increasingly harder equations but also doing so with much better efficiency. Computers made it possible to reach solutions that otherwise would have been impractical or impossible to obtain using analytical methods, making them indispensable tools for these types of problems.

2.3 Solver with current variable

When it comes to a current-based analog computer all of the blocks necessary for it to solve the problem at hand work with differential currents, both inputs and outputs, which makes it so that the line capacitance has less of an effect on the loaded ports. Contrary to a voltage based circuit where there is a need for a summing block, in this case by creating a node, two or more wires connected to each other, currents get added also it is possible to invert or subtract the signal just by cross-coupling. Despite having a lot of circuitry to operate in the desired way, current-based analog computers can still have a low power consumption.

By working with currents it is necessary to use an additional block, the fan-out block, which replicates the current as many times as needed. In spite of that because it is no more than a simple current mirror it isn't difficult to design. Unfortunately these blocks do consume significant power and are also known to hinder the accuracy of the whole solver.

Due to this type of architecture for a chip, which constitutes of blocks, these can be added or removed as the problem being solved requires, this makes for a design that is highly scalable to the needs of each individual problem.

Although it is not the focus of this dissertation, this architecture needs a digital controller, which is needed to program the operating modes, establish the initial conditions of the integrators and to configure the paths of each signal, also it is needed to do any calibration necessary and read the results outputted by the device.

2.3.1 Integrator Blocks

Integrators, Figure 2.3, are a pretty well known type of circuit in the industry as they are a key piece of this architecture that allows it to solve the differential equations which represent most of the problems being solved these days. Their output is subject to the changes of amplitude of the input signal and it's duration, that means that they contain the current approximation to the solution in a certain point in time.

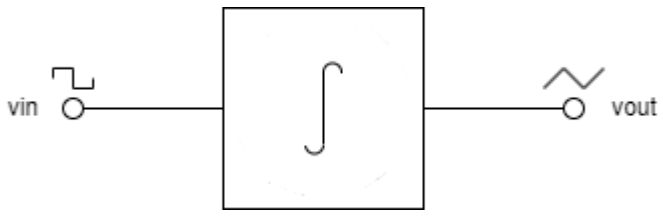


Figure 2.3: Integrator block

It starts off by being charged with an initial guess of the iterative method then by releasing it allows for the value of the output to deviate from the initial value and converge to a solution that satisfies the set of equations that defines the problem.

By integrating the input current that is supplied by the input current mirror devices, which make it possible to scale such current to a desirable value, it obtains the voltage at the capacitor which itself is then converted to a differential current by an Operational Transconductance Amplifier (OTA), making it possible for the other blocks to make their operations if required.

In this block there is a presence of some digital interface where it makes it possible to establish initial conditions of the integrator and it also makes it possible to program different gains for the current-mirrors although these having to be always reciprocal of each other.

2.3.2 Fan-out Blocks

Fan-out blocks are used in this architecture to copy and replicate an input current and output it to the other blocks. It is no more than a simple current-mirror device that can be programmed to have either unity gain or non-unity depending on the relation between the sizes of the transistors used.

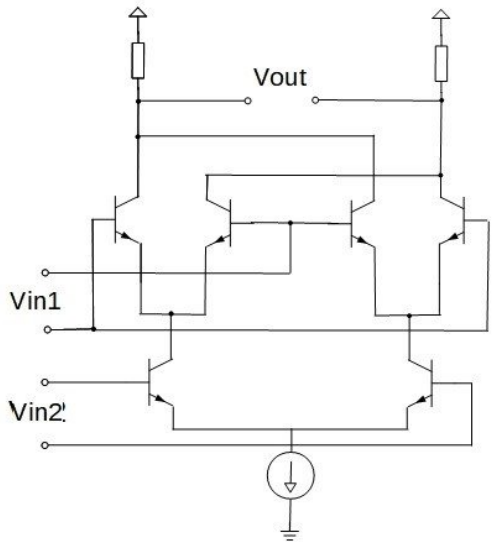
In spite of it being a relatively simple circuit it is important that the design is well implemented because it's possible to slow down the circuit just by increasing the input resistance.

2.3.3 Variable Gain Amplifiers/Multiplier Blocks

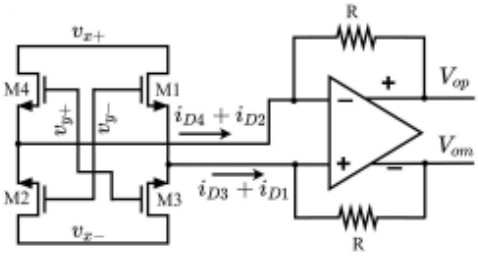
At each side of the core of this block, that can be either a VGA or a multiplier, there is a current-mirror, much like the integrator block, but unlike that block the gains of both input and output blocks are not reciprocal to each other, these also allow for the block to have multiple signal ranges.

This is also a digitally controlled block since it needs to be configured to be either a VGA or a multiplier, that's made possible by a control signal that depending on it's value alters the connections of the transistors that make this block, by doing so it changes the circuits characteristic and therefore the operation that it performs on the input.

In the case of multipliers these used to be implemented using gilbert cell type circuits, Figure 2.4a, when bipolar technologies were in use but when this typology of multiplier was implemented using CMOS transistors a limitation in accuracy and linear range were noticed [21], so nowadays designs that provide better results in these characteristics like a conductance multiplier seen in [22–24], Figure 2.4b.



(a) Example of a gilbert cell multiplier.



(b) Example of a conductance multiplier extracted from [25].

Figure 2.4: Comparison between the bipolar gilbert cell multiplier and a CMOS conductance multiplier.

2.4 Solver with voltage variable

In a voltage-based analog computer discrete spacial points and the differences in voltages are what define a problem's equation set, the electronic components of such computer produce a set of voltages that vary in time by being interconnected in such a way, making this set proportional to the modeled equations at each spatial point. This makes it so that the relation between the only variable in these equations which is time and computational time a unitary difference.

Contrary to a current-based architecture here there is an inherent need for a summing block to perform the addition which is the main operation used to solve these problems, to make this operation accurate this type of model must rely on summing currents at a virtual ground, which requires an opamp with high gain and Gain Bandwidth Product (GBW), resulting in a large-area, power hungry device.

These computers mostly tend to use switched-capacitor circuits due to the fact that these, compared to resistive circuits, are inherently more accurate, this is explained by that in CMOS processes a capacitor is easier to create with a better tolerance contrary to resistors. This enables this type of circuit to be highly adaptive, just by adjusting the ratios between these capacitors it is possible to adjust gains and create different transfer functions.

To make the computation of the equations that define the problem possible these computers rely on methods, described in the following section. By defining these equations it is possible to design the analog circuits of both the internal and boundary modules that constitute this architecture accordingly.

Just like in the current-based approach to an analog computer here there is also the need for digital interface, besides making it possible to convert the analog signal to digital so that it can be read and analyzed it also enables for the boundary modules to be each software programmable.

2.4.1 Methods

As said before the design of the internal and boundary modules that compose the analog computer is made with an update equation that results from the application of one of the methods to the set of differential equations that describe the problem. By connecting the resulting analog circuits systolic array that produces the result of this equation for a given spatial grid. Two main methods were found to be the ones that are most used in this scenario, the continuous-time in Laplace domain and the all-pass delay approximation [11, 26, 27].

In most of the problems from those areas mention earlier being described by differential equations that include time and space dimensions these methods will convert these equations into a suitable update equation that be converted to an analog circuit. So when it comes to the time dimension the first method applies the Laplace transform, but for the partial differential equations in the spatial dimension the discrete finite differences are applied to approximate these. Which results in an update equation that is both time-continuous but also spatially-discrete. By realizing this equation in an analog circuit it's possible to compute the continuous-time solution to the problem. The accuracy of such solution is also proportional to the order of the equation, which means the higher the order the more accurate the solution to the problem will be.

Lastly with the second method mentioned previously which is an improvement over the finite difference time domain method. In this method analog all-pass filter are used to create a continuous-time delay operator which will be used as a way to make up for the propagation delays caused by circuit elements. Making the main difference between this new method and the old that the time variable remains continuous and only the spatial is then discretized. Because of the compensations mentioned, an analog computer using this method will be able to achieve a higher bandwidth.

As these methods will end up allowing for the update equation to be converted to analog circuits that constitute the internal and boundary modules that are designed and connected in such a way as to produce a solution to the problem. By modifying the way the modules are connected it is possible to establish different boundary conditions.

2.4.2 Internal and Boundary Modules

Internal modules are made from opamps and all-pass filters, as said before by configuring the capacitor ratios it is possible to tune the signal gains with higher accuracy contrary to if it was made using resistors, all the while having less useful bandwidth making this a necessary trade-off because in analog computers the accuracy is privileged. It is also possible to avoid the accumulation of random Direct Current (DC) offsets in computation which results of the ac-coupled structure. Here the opamp accounts for the operations of addition or subtraction and by programming the gains of these opamps it is possible to achieve different configurations of coefficients of the non-linear equations.

As these modules function in parallel it means that the solution is being calculated at the same time for each position of the spatial grid. Also for each position of the problem's spatial grid it means that the input of the module is the output of the neighbour at the previous position and the output is the input of the next module.

The boundary module can vary depending on the boundary conditions that apply to the problem being solved. It can consist of an internal module but with extra opamps for additional operations with the boundary conditions or employ only a part of the used internal modules because an external excitation is applied and then an extra summing operation opamp is used. It can also be just the input of the first internal module connected to ground or an arbitrary voltage waveform.

3

State-of-Art

In this chapter the most relevant architectures to designing an analog computer will be looked at, it will go more in-depth to different approaches taken to solve each individual problem that they are targeting. From research, several designs were used and their basic principles were discussed in the previous chapter, in here the focus will be to analyze their results and conclusions.

As previously said analog computers are much faster at solving a single problem which they were designed to compared to a digital computer, this will be proven by the end of this chapter when the comparisons are presented. Also, because all of the problems solved are different from each other and create designs with different components and conditions, it will be possible to make a sensible comparison between them.

This will compare three different architectures of voltage-based analog computers and one using a current mode circuit with several types of digital computers from micro controllers to those based on Central Processing Unit (CPU), Graphics Processing Unit (GPU) or Field Programmable Gate Array (FPGA). This comparison will mostly be based on the time that these computers took to solve the problem of each approach and the accuracy that each solver was able to achieve.

3.1 Current-based solver

As said before analog computers usually rely on digital computers to operate, normally so that they can become more versatile. The approach to the analog solver that will be described here heavily depends on a digital interface to make it highly programmable to solve more than one problem, this is achievable by changing the initial conditions of the analog blocks and by altering the values of certain components.

In the design of these analog blocks it is important to realise that the time that takes to compute a solution is depending on the maximum computing speed of this computer. This solution time is a scalable portion of the interval of interest in a physical problem, this scaling factor is dependent on the values of the components of the circuit, mostly from the integrator's gain. By scaling these components of the integrator appropriately it

is possible to obtain different values for the maximum frequency of the computer which represents it's maximum speed.

One of the advantages that is made possible by this hybrid architecture, Figure 3.1, is the initial calibration of all the analog and mixed signal blocks, which is made by Digital to Analog Converter (DAC) that are placed in each block that by injecting differential currents at the startup of these blocks, allows for them to minimize the DC offsets [3]. The result of this calibration process, which is done individually to each block by a microcontroller running an algorithm to find the optimal code, is an error in the solution that is almost an order of magnitude smaller than when compared to no calibration.

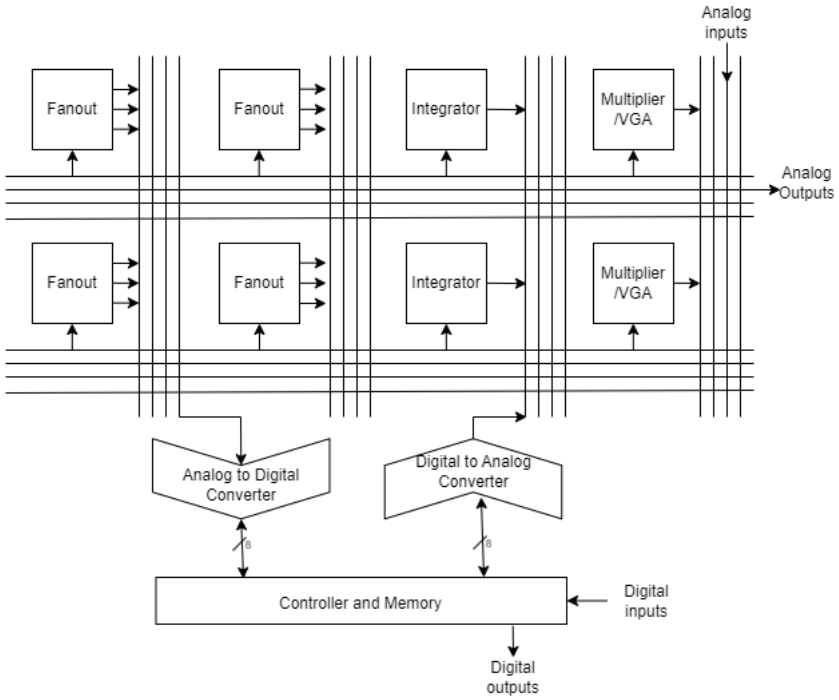


Figure 3.1: Current-based architecture of an analog computer, adapted from [3]

This approach used an integrator, Figure 3.2 ,with a gain that when facing device mismatches it isn't as sensitive as design in a previous work [28]. The current that is mirrored and then injected in the integration capacitor is produced by a class AB current mirror, this circuit's ratio can be set by changing the width of the mirroring transistors and to reduce the errors in the integration operation, which is done by increasing of the output impedance of the circuit which can be done using gain-boosting amplifiers. The integrator also has an output block of OTAs, which in this case are current-mirroring, and also a common-mode feedback OTA which by injecting the same current to both terminals of the integration capacitor maintains it's common-mode voltage. The previously mentioned DAC injects a current into a transimpedance amplifier to set the initial value of the capacitors voltage.

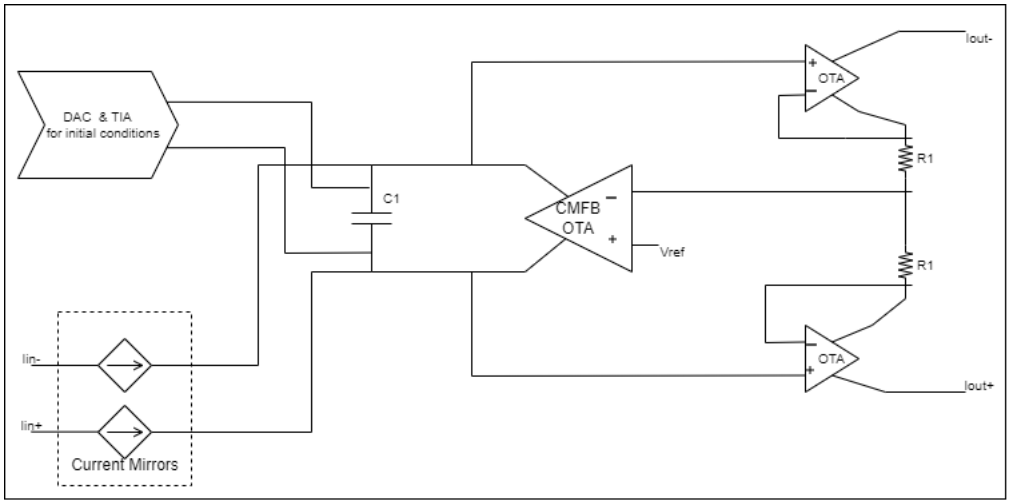


Figure 3.2: Integrator schematic

This architecture's digital components are implemented in continuous time which allow not only for low latency due the nonexistence of clocks, these devices only switch on when there is a need to complete a task which means it doesn't dissipate power in standby, also because it has no clock signal there is power dissipated to distribute this signal to all of the devices, and lastly it has a better computation accuracy due to the lack of aliasing. The use of this type of circuitry allows for more information to be carried out compared to conventional digital circuits, because it uses binary signals that are a function of time, which means the signal has a value continuous in time.

This digital side of the computer's architecture can either be used just for calibration and the setting of initial conditions and constantly checking if any blocks have reached saturation, such functionalities are shown in Figure 3.3. But also because of the Analog to Digital Converter (ADC) it can also take a part in the solving of the equation by reading the analog values and converting them to digital which then are used by the memory to look for the non-linear values, which then are converted by the DACs and sent to the analog blocks by the fanout devices.

Due to this architecture being highly modular and re-configurable to solve many types of problems using non-linear ordinary differential equations, two different types of problems, both a coupled mass-spring problem, Figure 3.4, and a Van de Pol equations 3.1, were solved to test this computer and both of this results were compared to a Texas Instruments microcontroller of which the comparison is section 3.3.

$$\dot{x_1} = x_2 \tag{3.1a}$$

$$\dot{x_2} = \mu(1 - x_1^2)x_2 - x_1 \tag{3.1b}$$

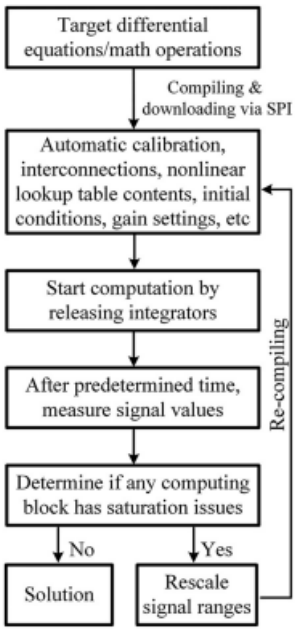


Figure 3.3: Functionalities of the digital side of this design, extracted from [3]

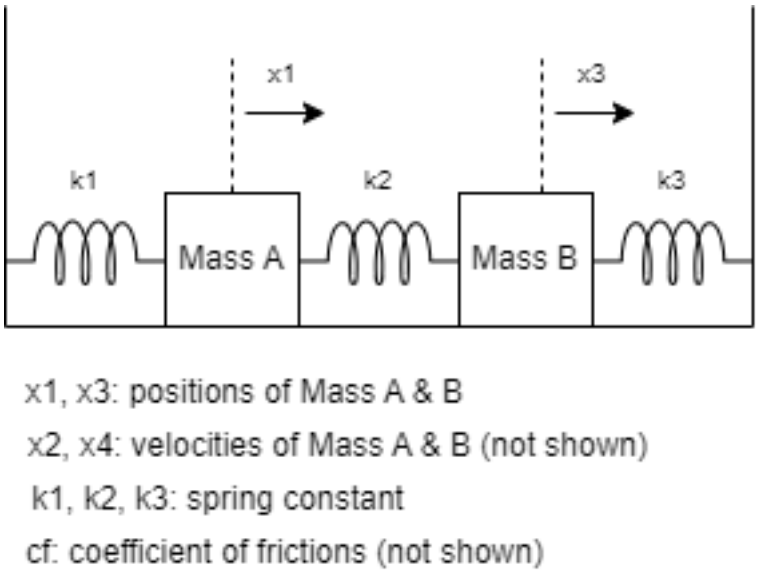


Figure 3.4: Example of a coupled mass-spring problem, adapted from [3]

3.2 Voltage-based solver

In this section three approaches to voltage-based analog computers with different target problems will be discussed and analyzed. First the equation that define the problem will be presented, then the method that converts this into an update equation is mentioned, so that the components needed to implement the modules are shown and explained, lastly to design the boundary modules their conditions and equations are presented.

3.2.1 One dimension wave equation solver

This approach took on the problem of solving a one dimension wave equation 3.2, which is relevant in the area of physics, but with several types of boundary conditions which would be controlled by an FPGA, this device also is responsible for the reading of the outputs and the respective post-processing. This architecture also relies on an additional FPGA that provides the input wave-forms to excite the system.

$$\frac{1}{c_0^2}\frac{\delta^2 E(x,t)}{\delta t^2} = \frac{\delta^2 E(x,t)}{\delta x^2} \tag{3.2}$$

The method used to design the circuits that describe the problem's equation will be the all-pass delay approximation because as said previously it can more easily compensate for errors in the circuits implementation, like propagation delays [26]. By approximating the both of the derivatives of the electric field, in time and space, is allows for an update equation 3.3 to be reached which is then used to design the internal modules that allow for the wave equation to be solved. The time delay that is present in this update equation will be represented by an all-pass filter that has an equivalent group delay.

$$E(i,t) = K^2[E(i+1,t-\tau) + E(i-1,t-\tau)] + 2(1-K^2)E(i,t-\tau) - E(i,t-2\tau) \tag{3.3}$$

In this type of architecture the output of an internal module is equivalent to the value of the electric field at that spatial position, so by interconnecting all of the necessary modules, as shown in Figure 3.5, it allows for the solution to the problem to be computed over the spatial grid.

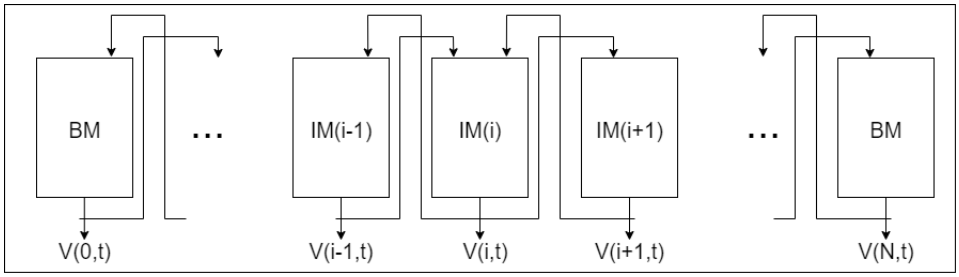


Figure 3.5: Design of an architecture implemented with IM and BM

These also require boundary modules to compute the solution which depending on the type of boundary condition, Dirichlet, Neumann or radiation, which in the case of the first one implies injecting a DC voltage to last of the internal modules, as for the second there is the need to apply the finite difference approximation due to the spatial derivative being equal to zero at the right boundary which results in a different update equation at that position, leading to a design that uses an internal module with both of its inputs connected to each other. Lastly the radiation boundary the update equation needs to satisfy a condition 3.4 that the sum of both fisrt order derivatives of the electric field in space and time are equal to zero. Then the update equation is obtained by applying the centered finite differences to both space and time dimensions which leads to a circuit that

is similar to an internal module but with only an input for the output of the previous module.

$$\frac{1}{c_0}\frac{\delta E(x,t)}{\delta t}\|_{x=N\Delta x} + \frac{\delta E(x,t)}{\delta x}\|_{x=N\Delta x} = 0 \tag{3.4}$$

This circuit was designed using a mainstream CMOS technology from TSMC that uses a 180nm node, it implemented 15 internal modules that all have the same design, 2 boundary modules, one that realize either a Neumann or a Dirichlet boundary condition and the other is used to implement the radiation boundary condition, another module is used, that is similar to an internal module, to control which boundary condition is used this acts on a control signal sent from the digital interface.

The internal modules' design is shown in Figure 3.6, it implements an opamp that needs to have a high GBW which is required for acceptable accuracy in continuous-time computations, is also employs an all-pass filter that itself is a cascading of three first order all pass filters although these being separated by buffers to keep the loading of each other low and finally a subtractor that is used which was designed so that the overall gain of the filter is unity. They also include arrays of capacitors so that it is possible to compensate for power, voltage and temperature variations, parasitics and mismatches and a resistor to provide DC feedback to the opamp. The use of opamps in this architecture introduces propagation delays which will have to be taken to account when designing the delay of the all-pass filter.

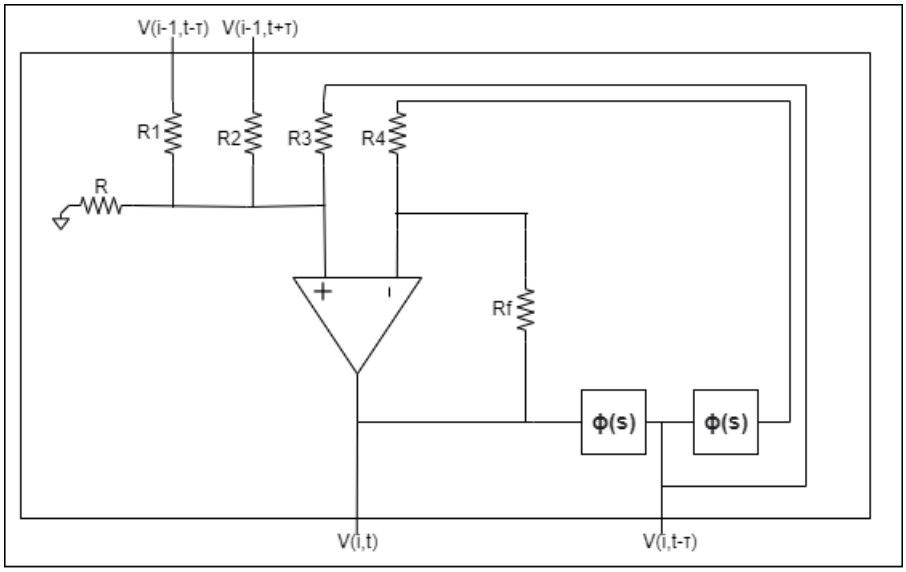


Figure 3.6: Schematic of an internal module solving a one dimension wave equation, adapted from [26]

The bias voltages of the circuit were set using an Simultaneous Perturbation Stochastic Approximation (SPSA) optimization algorithm which starts off by setting an initial value for this voltage that were determined by simulations and by using an iterative process to convert these bias voltages by analyzing the function that represents the losses of the circuit.

3.2.2 Transmission line problem solver

This approached is aimed at solving one dimension wave equations that mostly describe eletromagnetic problems by using an analog computing algorithm that applies the finite-difference time-domain method to this wave equation in order to convert it to a simple update equation [29]. This design is implemented using switched-capacitor circuits for the reasons stated in the previous chapter.

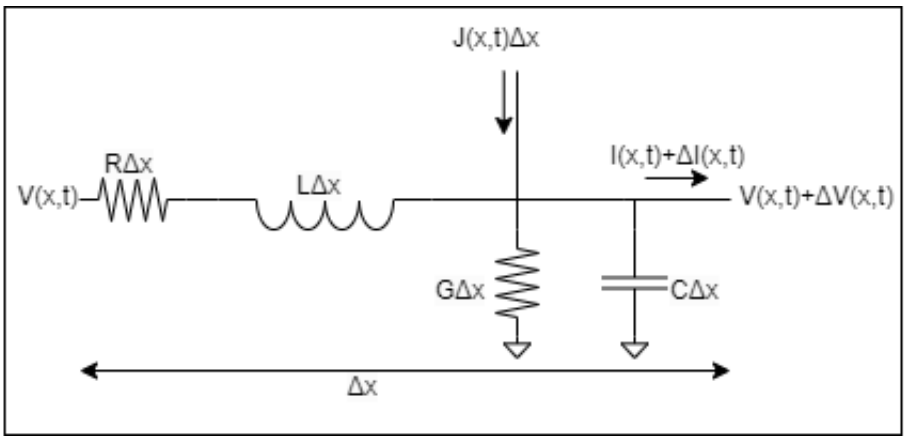


Figure 3.7: Model of a transmission line problem, adapted from [29]

An electrical model of a wave equation 3.5 can be described by a transmission line, this means a one dimension wave equation can be derived from a transmission line, which is represented in Figure 3.7. From the Kirschoff's voltage and current laws it is possible to get the difference equations of current and voltage which after some approximations and applying the finite-difference time-domain method it is possible to obtain the update equation 3.6 that can then be modeled into the internal modules of the analog computers.

$$\frac{\delta^2 W(x,t)}{\delta t^2} = c^2 \frac{\delta^2 W(x,t)}{\delta x^2} \tag{3.5}$$

$$k_1^2 W_d(n_x, n_t+1) = W_d(n_x+1, n_t)+W_d(n_x-1, n_t)+2(k_1^2-1)W_d(n_x, n_t)-k_1^2 W_d(n_x, n_t-1) \tag{3.6}$$

Just like in the previous approach, to solve a differential equation it is necessary to establish boundaries, therefore there is the need to implement boundary modules, which brings back the same conditions as before, Dirichlet, Neumann and radiation. The first condition that in this case implies a termination to the transmission line, impedance equal to zero, makes for module that is implemented using an internal module with one of it's inputs shorted to ground. A Neumann condition in this scenario is equal to open circuit at the termination of the transmission line, that means an impedance that is infinite, which can be realized by connecting the two inputs of an internal modules and connecting them to the output of the adjacent module. Lastly to the radiation boundary is designed using a different circuit because it's gain input values differ from the ones of an internal modules, this boundary condition 3.7 represents a matched impedance termination to the line, the equation that represents this module satisfies that the sum of both derivatives, in time and space, are equal to zero and is obtained after a second order finite difference approximation is applied.

$$\frac{1}{c}\frac{\delta W(x,t)}{\delta t}\|_{x=N_x\Delta x} + \frac{\delta W(x,t)}{\delta x}\|_{x=N_x\Delta x} = 0 \tag{3.7}$$

As previously said this analog computer was designed using switched-capacitor circuits to implement its internal modules which is implies an acceptable trade-off of useful bandwidth for accuracy of the solution. In the design of these modules, shown in Figure 3.8, it employs five stages, the first consists of a summing amplifier with auto-zeroing capabilities, the following four stages are each composed of a sample and hold buffer. The auto-zeroing comes from the ability of the amplifier to cancel out any offset that might have come from the previous stages, these amplifiers also come with the possibility of programmable capacitors to simulate different wave propagation speeds. In order to set-up the desired feedback voltage these amplifiers also implement a reset switch with it's own phase that is only periodically triggered so that the common mode feedback circuit can set all of the DC voltages. The sample and hold buffer is used to create the time delay that the mathematical problem needs, this type of buffer also improves settling time and linearity.

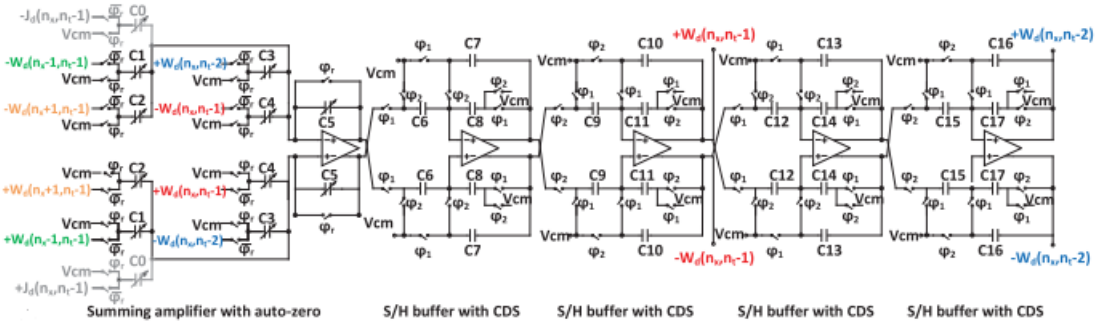


Figure 3.8: Design of an internal module for the transmission line problem, extracted from [29]

In this architecture, delta-sigma modulators were also used to convert the analog signals to digital so that these can be processed and read, these allow for a small number of output pins and for the digital signal to be easily converted to back to analog through low pass filter, by applying single-bit quantization. All of the integrators that realize the modulator also use the same amplifier as the internal modules, although the fisrt integrator uses the correlated double sampling technique to minimize the noise and compensate the gain of the amplifier, the second and third do not use this method to simplify their design but instead their gain can be programmable because of their separate input and DAC capacitors.

3.2.3 Variable area duct problem solver

Lastly, in this last approach to an analog computer that solves differential equation it took the previous designs and extended them to solve non-linear differential equations since these are used to describe more complex problems in the physics department, which makes these even harder on even the most powerful digital computers. This being a hybrid computer which has an analog core that will compute the solution to the equation that is programmed by the digital side, this is done by System on a Chip (SoC) that in conjunction with ADCs can alter the initial and boundary conditions.

Since designing an analog computer that can solve any type of non-linear differential equation is virtually an impossible task, this approach targets equations that are hyperbolic in nature also focusing more on conservative systems that are one dimensional, which means it's variables are space and time. To obtain an update equation that can then be implemented into a circuit there is the need to apply a method to the differential equation, as previously stated, this approach focused on a numerical method, the MacCormack's method [30, 31], that by being a more simple scheme, a second-order instead of one of a higher order, which results in a simpler circuit, on the downside of having a little less accuracy compared to those of a higher order [32, 33]. This is a spatially-discrete time-continuous method that relies on the all pass delay approximation to implement the delays needed for the two computational steps, predictor and corrector.

This is will result in an internal module, Figure 3.9 that is highly flexible in the amount of problems it can solve, by having programmable gains in the non-linear function blocks that are present in these modules allows for the tuning of these gains or even adding or removing some of the terms to represent different differential equations. By limiting these non-linearities to second order, it means the analog circuits needed to implement this module are only multipliers, opamps and the programmable gains that were mentioned [25]. Likewise to the previous approaches by varying the capacitor ratios it is possible to tune these gains and for the addition or removal of functions digitally controlled switches are employed.

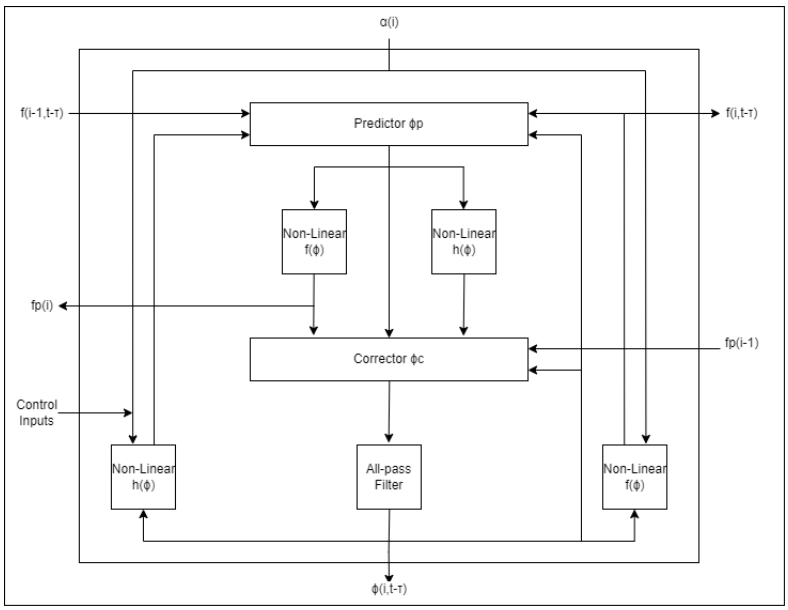


Figure 3.9: Design of an internal module for the variable duct problem, adapted from [25]

The circuits of this approach were designed to be fully differential, this means that each non-linear solution of the partial differential equations is proportional to the time-varying differential voltages produced by the analog computer. This choice in architecture when compared to single ended reflects in a dynamic range that is 3dBs higher and results in a lower offset voltage and even-harmonic distortion [25].

Both the all-pass filter and the multipliers will both be using the same opamp, that in this case is two-stage fully differential opamp with a common mode feedback circuit, in a transimpedance amplifier setup as their last stage. In the case of the all-pass filter it uses three first order RC filters in cascade isolated by a buffer, being the capacitor ratios adjustable to tune the group delay and the gain of this system. As for the multiplier it uses a CMOS conductance design [22–24] with four transistors that form a multiplying quad, by operating in the triode region and being long channel devices they improve the linear range of the multiplier [34].

Contrary to the other approaches in the voltage based architectures, this only simulates one boundary condition for each side of the spatial vector, this being that the left boundary

in only an arbitrary voltage value injected into a summing opamp that then by employing the one of the non-linear function blocks is able to compute the spatial solution for the other term of the equation. For the right boundary an internal module is used but with the addition of four summing and scaling opamps. Due to this being a problem of a varying area duct the coefficients used in the non-linear blocks were determined by the characteristics of the duct itself and the exit section that corresponds to the boundary values.

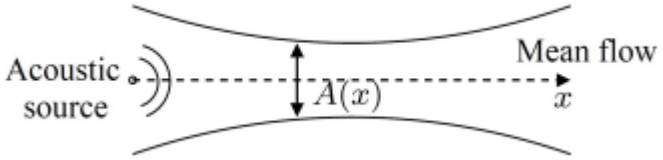


Figure 3.10: Variable area duct problem, extracted from [25]

3.3 Solver results

In this section the results of each approach to an analog computer targeted at solving differential equations will be shown. As previously said there are two metrics that can be used to first compare an analog solver to an equivalent digital system, the first being the time it took for both systems to compute the solution to the problem and second the relative accuracy of the solution from the analog computer compared with the solution from the digital solution.

In the subject of accuracy two different metrics were used to describe how accurate the analog solution was compared to the digital one. The first one being the normalized mean squared difference, this is obtained by first calculating the mean squared difference between the analog solution and the finite difference time domain digital solution at all the spatial grid points of the problem and then normalizing it to get the results in a decibel scale. The second being the normalized root mean squared which is given by the root mean square of both analog and the ideal solution and then converted to a relative of the latter. With this in mind the results were split into two tables, table 3.1 for the results using Root Mean Squared (RMS) scale which are from the current based analog computer and table 3.2 for the results presented in the normalised Mean Square Difference (MSD) scale.

System	NRMS (%)
First approach solving a coupled mass-spring system problem	4.7
First approach solving a second order Van der Pol equation	4.6

Table 3.1: Accuracy results of the current-based analog computer

System	γ (dB)
First approach with a radiation boundary	-20 to -10
Second approach without calibration with a radiation boundary	-25 to -7
Second approach with calibration with a radiation boundary	-20 to -10
Third approach with a continuously varying duct	-27 to -7
Third approach with a notch like restriction near left boundary of the duct	-32 to -14

Table 3.2: Accuracy results of each voltage-based analog computer

Secondly the results of the solution times to the computation of each approaches problem will be presented in table 3.3, it is important to remember that because analog solvers compute the solution in real time, whereas the digital computers implemented with CPUs and GPUs in order to compute a single temporal frame of the update equation take many clock cycles, so there is the need to ensure that the comparison between both systems is fair, to enable this there is the need to guarantee that the input signal to both systems is the same and in case there is a time step this also needs to be matched to the analog computer's, also the boundary conditions must be programmed in the same way as the analog solver.

System	Speed-up time
Current-based approach solving a coupled mass-spring system problem	128x
Current-based approach solving a second order Van der Pol equation	154x
First voltage-based approach	2.8x-420x
Second voltage-based approach	1.29x-21.1x
Third voltage-based approach	140x-205x

Table 3.3: Difference in speed of the analog to digital computer

3.4 Solver comparison and discussion

In this section the results seen from each approach in 3.3 will be compared so that a sensible comparison can be made between each technique that was used in the expectation that a conclusion can be taken about each approaches performance.

From these results it is possible to realize that as expected because each of the voltage-based approaches are an incremental improvement over the other slightly older, they also get better results than the previous ones, being that the last approach reported in 3.2.3 has the best results in the time that it took to compute the solution compared to it's digital counterpart, also this approach solves non-linear differential equations which are more complex than linear ones. The current-based analog computer also presents comparable results to the previously mentioned voltage-based approach and also has the same capability of solving non-linear equations. However one way to decide which design would be more suitable for the application needed could be through their power usage, where the current-based design has a far less power hungry architecture at only 1.2mW

compared to the other approach consuming 3W.

Taking these results into account it can be quickly realized that an analog computer, as expected and described in section 2, is a faster and more power efficient way to solve a differential equation compared to a digital computer, however the former has a limit in how many different equations it can be configured to solve, as the latter has virtually unlimited capabilities as long as time isn't a defining factor.

With all of this being said it seems that the current-based solver seems more suitable to be implemented in this type of work since there is the possibility to start with a simple equation that doesn't take many of the mentioned blocks to compute and work in the way of increasing the difficulty of said equations and the amount of blocks needed to design and implement. This also leaves room for future work in the area of building an equivalent voltage-based system that implements the same equations so that a proper comparison can be made using the same technology and conditions.

4

Proposed circuit and analysis

In this chapter which includes the theoretical part of this thesis will be explained so that in the next chapter the results of the simulations can be showcased and discussed. This will be done by in a first stage analyzing and designing the components that are required to solve a differential equation, and then in a later stage the proposed circuit and problem will be analyzed.

When it comes to the part of solving a differential equation, it will start of by analyzing a generalized linear differential equation and the circuit that defines it that were taken from [35] to understand how this type of circuit would behave and how it relates to the defined equation. Then a real world problem that uses a similar equation will be presented to show that solving this type of equation can be a relevant matter, and finally an effort will be made to make this circuit programmable so that it can solve more than one fixed linear differential equation.

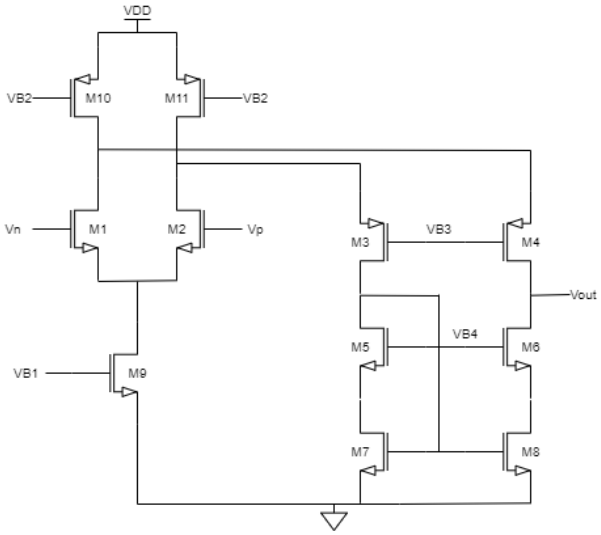
Lastly, since this is a thesis in Electronics engineering, the components that are necessary to make it possible to solve such equations will be discussed, such as the integrator which itself relies on an opamp. Therefore it will go in depth to the process of designing this amplifier and how itself was configured as an integrator as well as the challenges faced when using a real amplifier as an integrator.

4.1 Amplifier

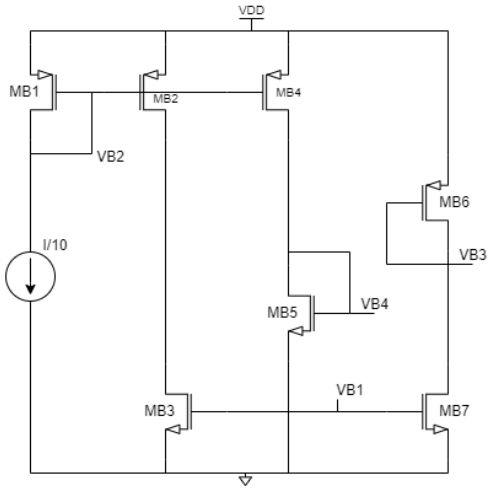
A good opamp is going to be the foundation of this whole work, due to the fact that the integrator block, the one that allows for the solving of differential equations, requires an opamp in it's core. However it will also be used in two more common configurations, as a summing amplifier as well as a simple inverter montage.

From the research done before the start of this project some characteristics were found to be important for this amplifier. Most importantly as the computing bandwidth, which is maximum frequency that allows for accurate results of the processed signals, is set to 20kHz, therefore the bandwidth of the circuit has to be 100 times greater to maintain acceptable gain and phase in the computing frequency range.

For this thesis a simple, single supply, Folded Cascode Amplifier (FCA), Figure 4.1, was chosen because it allows for a high gain to be reached, but most importantly it's a simple design that allows for a wide bandwidth to be achieved, has a good output swing which allows for the opamp to be able to handle stronger input signals without distortion and lastly because it has good stability as long as it is well biased.



(a) Amplifier circuit.



(b) Biasing circuit.

Figure 4.1: Amplifier circuit and it's biasing circuit.

In order to design this opamp there are a few steps that must be taken, first the characteristics of the opamp that are relevant to this work must be defined so that when designing it there is a target baseline that needs to be reached. Secondly the equations that define these must be obtained from analyzing the circuit. After that, with the help of a Mathcad sheet with the characteristics for the technology that's going to be used, the parameters of each transistor are going to be tuned to achieve the established targets. And finally, in the next chapter, the simulations to verify these results are going to be shown.

4.1.1 Required characteristics

As previously mentioned it is important to establish the characteristics of the amplifier's circuit so that when tuning the parameters of the transistors later on in the next section a baseline can be had as to which values must be achieved. The following list represents which values for each characteristic were thought of for the circuit.

- $A_v \geq 60dB$:
- $GBW \approx 50MHz$;
- Second pole frequency: $fp_2 \geq 100MHz$;
- Output swing: $OS \geq 550mV$.

There was also the need to maintain the vdsats between 50mV and 150mV to guarantee that the transistors were in moderate inversion, while also assuring that the area of each transistor was under 1 micro meter squared due to matching reasons. As an additional objective it was established that the power dissipation should be as low as possible as well as the total die area.

4.1.2 Relevant equations

Since this is a differential amplifier, which means it has two sides that are symmetrical to each other, all of the following equations that are going to be taken from the analysis of the circuit can be done just by looking at one of the halves of the amplifier, which explains why only some of the transistors will mentioned in the equations and mathcad graphs.

Firstly the most important equation that is going to obtained by analysing the circuit is going to be the DC Gain of the FCA circuit, which is the transconductance of the input transistors multiplied by the output impedance of the montage. At the output node, there is M6 cascoded with M8 and M4 cascoded with the parallel of M1 and M10, which means the output impedance is the parallel of two impedances. Therefore the impedance seen below and above the output node is given by 4.1 and 4.2, respectively.

$$ro_n = \frac{gm_6}{gds_6} * rds_8 \tag{4.1}$$

$$ro_p = \frac{gm_4}{gds_4} * (rds_1 // rds_{10}) \tag{4.2}$$

$$A_v = (B_E * gm_1) * (ro_p // ro_n) \tag{4.3}$$

Secondly there is the need to reach the expression for the GBW, which is given by the DC Gain multiplied by the output pole's frequency, which requires that the equation of the output capacitance is obtained, as seen in 4.4.

$$C_o = C_L + C_{db6} + C_{db4} + C_{gd6} + C_{gd4} \tag{4.4}$$

With the output capacitance calculated it is possible to obtain the equation for the GBW, as seen in 4.5.

$$GBW = A_v * w_{po} = \frac{B_E * gm_1}{C_o} \tag{4.5}$$

Next there is the input-referred noise of the amplifier circuit, since the transistors in common-gate do not contribute to the noise they won't be taken into consideration for this equation. With this being said the only transistors that add noise to the circuit are the M11 and M8, therefore it is possible to achieve the following equation 4.6.

$$ENF = (2 + \frac{gm_{11}}{BE * gm_1} + \frac{gm_8}{BE * gm_1}) \tag{4.6}$$

In order to guarantee stability in a unity gain closed loop configuration of this opamp there is the need to obtain the equation for the second pole, so that it's possible to make this pole sit at least twice the frequency of the GBW. The expression 4.9 for this pole is obtained from the node that connects M1, M4 and M11 where there is the need to get this node's capacitance 4.7 and impedance 4.8 expressions.

$$C_{fp2} = C_{db11} + C_{db1} + C_{gd11} + C_{gd1} + C_{gs4} + C_{sb4} \tag{4.7}$$

$$R_{fp2} \approx \frac{1}{gm_4} \tag{4.8}$$

$$fp_2 = \frac{1}{2 * \pi * R_{fp2} * C_{fp2}} \tag{4.9}$$

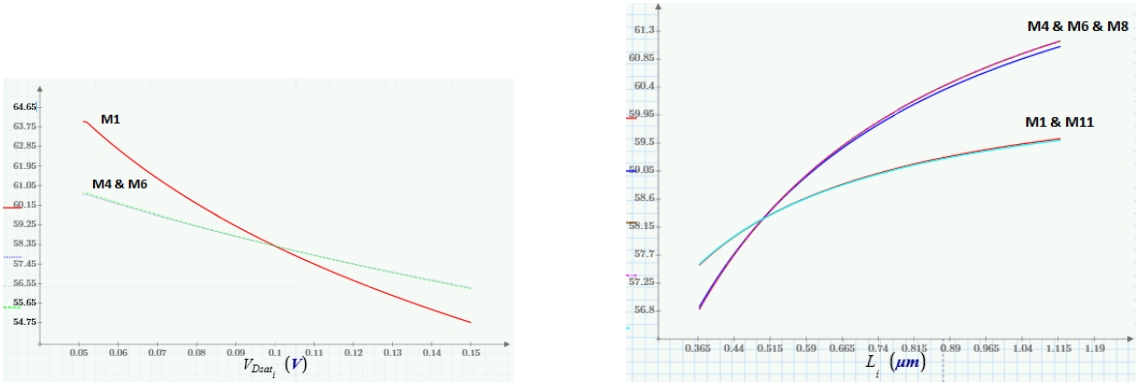
Lastly there is the equation for the output swing, 4.10, where to obtain it, the vdsats of the transistors where the signal goes through them towards VDD and towards VSS need to be summed, all while also adding a margin of 200mV. This is important to guarantee that all the transistors stay in their active zone.

$$OS = VDD - (V_{dsat4} + V_{dsat6} + V_{dsat8} + V_{dsat11} + 200mV) \tag{4.10}$$

4.1.3 Mathcad and Final design

After taking the equations from the analysis of the circuit, these were written in the Mathcad sheet that has the characteristics of the transistors so that by analysing which of these has more impact on each of the parameters of the circuit, it would be possible to configure the dimensions of the transistors in order to obtain the objectives that were set. The following images are the graphics that were taken from Mathcad that represent the relations that were previously mentioned.

On Figure 4.2 it is possible to analyze that all vdsat's have some impact on the gain of the amplifier, albeit the vdsat of transistor M1 has a bigger effect on this parameter. The same can be said about the transistor length, although now the transistors M1 and M11 have less impact on the gain of the opamp.

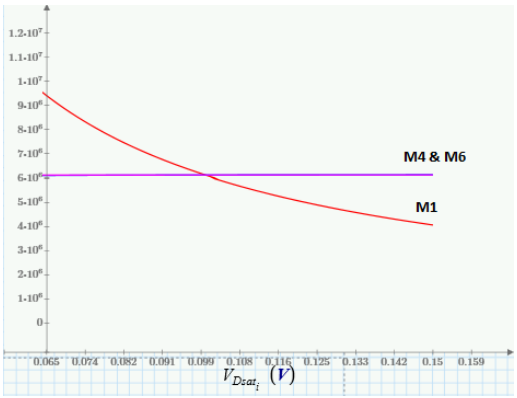


(a) How gain relates to vdsat.

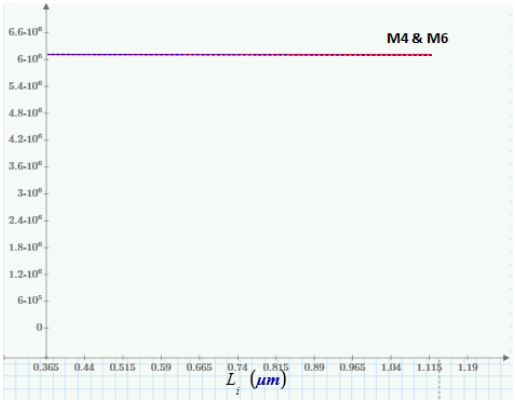
(b) How gain relates to transistor length.

Figure 4.2: Amplifier gain as a function of vdsat and transistor length.

From Figure 4.3 it is possible to conclude that the GBW is only impacted by the vdsat of transistor M1, and the length of the transistors has no effect on this characteristic of the circuit.



(a) How the gain bandwidth product relates to vdsat.



(b) How the gain bandwidth product relates to transistor length.

Figure 4.3: Gain Bandwidth Product as a function of vdsat and transistor length.

From the analysis of Figure 4.4 it is possible to come to the conclusion that the bigger the vdsat on transistor M1 the higher the noise figure is going to be, on the other hand both the transistors M8 and M10 have the opposite effect, which means the higher the vdsat the lower the noise figure is.

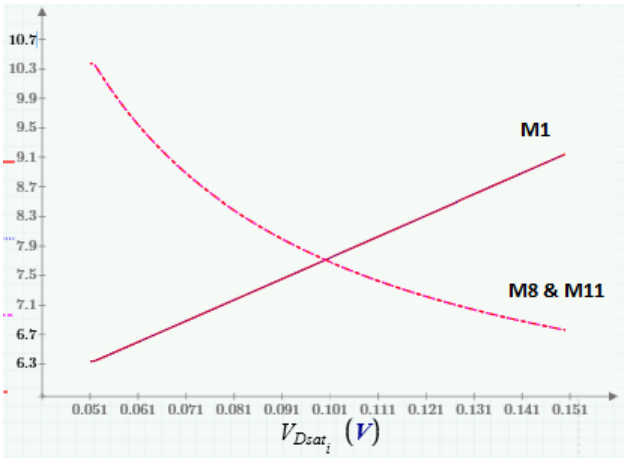
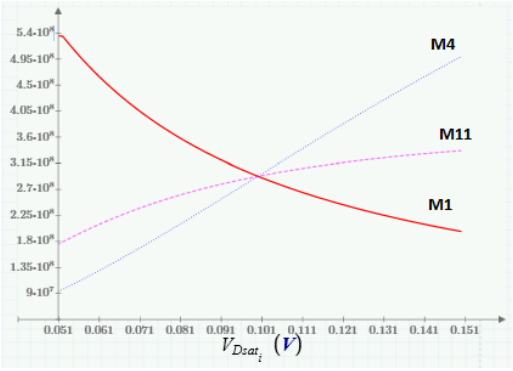
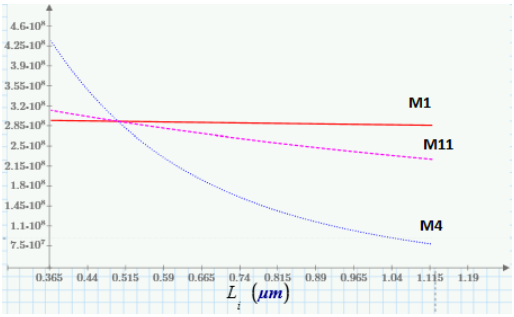


Figure 4.4: Noise as a function of vdsat

Lastly by analyzing Figure 4.5 it can be concluded that all the vdsats have some impact on the frequency of this second pole, vdsat4 and vdsat11 have a positive impact as they increase and vdsat1 has the opposite effect, lowering it as it's value gets higher. As for the length of transistor M1, it has no impact on this parameter, which can't be said for the length of M4 and M11, where the former has a small negative impact as it increases and the latter has an even more expressive negative effect as it increases.



(a) How the second pole frequency relates to vdsat.



(b) How the second pole frequency relates to transistor length.

Figure 4.5: Frequency of the second pole as a function of vdsat and transistor length.

Finally after tuning the values of these parameters it was possible to achieve the characteristics for the circuit presented in table 4.1, and as a result of the achievement of the former values, in the table 4.2 it is possible to see the width and length of each transistor that makes up this opamp.

$Gain\ (dB)$	$GBW\ (MHz)$	$fp2\ (MHz)$	$OS\ (V)$	ENF	$I_B\ (\mu A)$	$Pd\ (mW)$
62.9	50.1	105.6	0.6	4.46	236	0.68

Table 4.1: Circuit results after the tuning of the transistor parameters

	$M1$	$M4$	$M6$	$M8$	$M11$	$M9$	$MB1/2/4$	$MB3/7$	$MB5$	$MB6$
$L\ (\mu m)$	0.36	0.36	0.36	1	0.36	1	0.36	1	1.2	1.2
$W\ (\mu m)$	138	173	138	47	75	94	7.5	9.4	1.2	12.2

Table 4.2: Final sizing of the circuit transistors

Finally with all the equations taken and the circuit designed, all that is left to be done is for the characteristics that were achieved to be validated through simulations in cadence's software, this is going to be done in the next chapter.

4.2 Integrator analysis

As previously mentioned integrators take a key role when it comes to solving differential equations using analog circuits, this is made possible because is generates an output signal that is proportional to the integral of the input signal, by using the opamp that was previously analyzed set up with a capacitor in it's feedback loop.

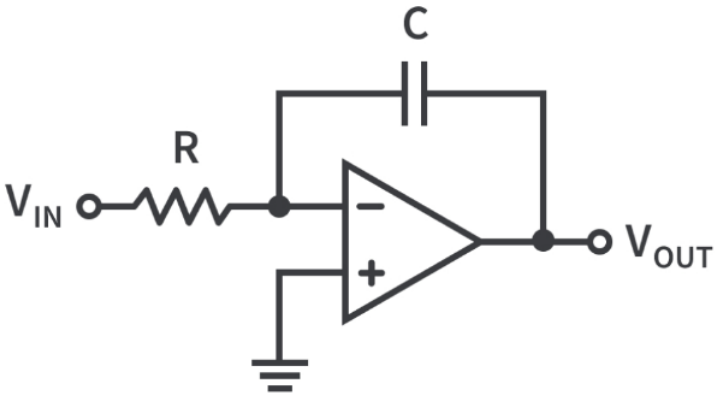


Figure 4.6: Opamp in an integrator montage

From analyzing Figure 4.6, equation 4.11 can be taken which describes the output as a function of the input.

$$V_{out} = -\frac{1}{RC}\int V_{in}dt \tag{4.11}$$

There is a problem however when it comes to using a real amplifier set up as an integrator, because it can bring reliability issues to the solution of the problem being solved. This is caused by the input offset voltage of the opamp, which unlike an ideal amplifier, the two input terminals don't have the same voltage, this means this small variation will cause for an error that builds over time. Also because the capacitor at low frequencies acts as an open circuit, which means that there is no feedback from the output to the input, making it so that the circuit behaves as a high gain inverting amplifier [36].

These problems can be solved by introducing a resistor in parallel with the capacitor in the feedback loop, resulting in the circuit seen in Figure 4.7.

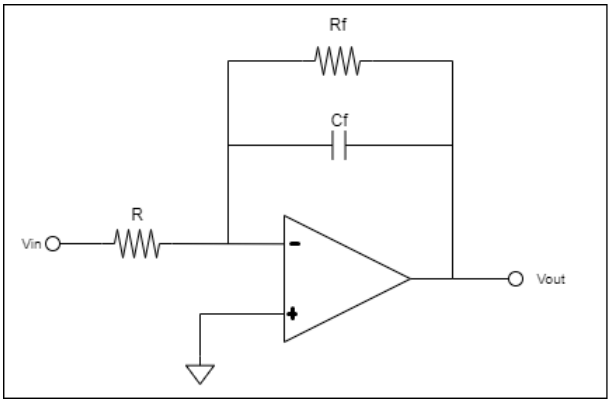


Figure 4.7: Opamp in an integrator montage with a feedback resistor

This means that at low frequency the circuit behaves as an inverting amplifier with a finite closed loop gain given by the expression 4.12. As the frequency increases and the capacitor charges this resistor gets shorted due to the effects of the capacitive reactance, thus reducing the gain on the circuit.

$$A_v = -\frac{R_f}{R} \tag{4.12}$$

To understand if the amplifier that was designed was working as intended as an integrator, two simple tests were conducted ahead of attempting to solve the actual problem of this thesis, the linear differential equation. To demonstrate this, two different wave signals will be introduced to the input of the integrator, a sine wave and a square wave. For the first signal the integration is quite simple and results in a cosine wave divided by a factor equal to its frequency, as shown in equation 4.13.

$$V_{out} = \frac{1}{RC}\int sen(\omega t)dt = \frac{1}{\omega RC}cos(\omega t) \tag{4.13}$$

For the square wave analyzing the integration is a bit different as it needs to be separated into two because of the two different voltage values that make this wave signal, as can be seen from equation 4.14.

$$\begin{cases} V_{in} = \pm V_1 \\ \\ V_{out} = \dfrac{1}{RC}\displaystyle\int_{t_1}^{t_2} \pm V_1 dt = \pm\frac{1}{RC}[t_2 - t_1]V_1 \end{cases} \tag{4.14}$$

4.3 Proposed circuit

Finally after analyzing the amplifier, the main component of this work that is going to give the possibility to build a circuit that solves a differential equation, and understanding how it can be set up as an integrator and what constraints are brought up by working with a non ideal amplifier, it is time to address what type of problem is going to be approached in this thesis.

This work will start off by analyzing a general linear differential equation and the circuit that describes it, so that the behavior of the circuit and how it relates to the equation can be well understood. Learning how the coefficients of the expression relate to the components of the circuit will be crucial so that it is possible to target a more specific and relevant equation with real world applications.

With this in mind there was an effort made to find an equation that was both relatable to the equation that was analyzed and had it's circuit already designed as well as had a relevant application, settling on the law of exponential decay equation. In a later stage, after analyzing how the two equations relate and the connections to the components of the circuit are established, the generalization of the circuit would be implemented by introducing switchable components.

4.3.1 Linear differential equation

Firstly a linear differential equation and the circuit that describes it are going to analyzed. It is called a linear equation because there are no products or higher powers of the unknown function or it's derivatives, therefore the circuit necessary to solve this equation is much simpler, being only necessary to use an amplifier in a summing montage and an integrator montage. With this in mind, equation 4.15 represents the example of a linear differential equation that is going to be solved.

$$Ay'(t) = -(By(t) + C) \tag{4.15}$$

When this equation is translated to an analog circuit aimed at solving it, it results in the the circuit seen in Figure 4.8.

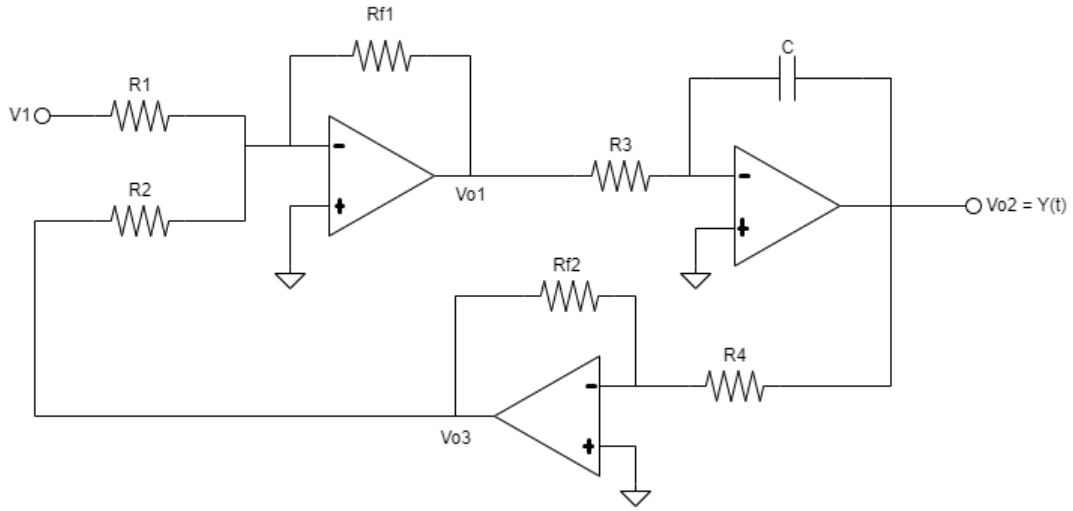


Figure 4.8: Circuit of a linear differential equation, extracted from [35]

Next there is the need to analyze this circuit so that a general idea of how it is supposed to behave through out its nodes. Starting with node V_{o2} which would be the solution to the equations which means it corresponds to the variable y. After that there is the feedback loop of the circuit that consists of an opamp in an inverting unity gain montage, which means in V_{o3} the signal would be a negative y. Then it comes into an inverting summing amplifier, where the feedback signal is initially added to an excitation signal, which is a switched DC voltage. Which results in equation 4.16 for node V_{o1}, where in this case V_1 will be considered as a negative source.

$$Vo_1 = By + CV_1 \tag{4.16}$$

As the last step in this circuit there is the opamp set up as an inverting integrator montage, which will integrate the signal as well as invert it, as equation 4.17 shows.

$$Vo_2 = \int -By(t) - C\,dt \tag{4.17}$$

With these equations in mind, a connection between these coefficients and the components of the circuit can be established, which would result in the expressions from 4.18. Coefficient A is missing since it can be related to the other two but in this case will always be considered with a value of 1.

$$\begin{cases} B = \dfrac{R_{f1}}{R2} * \dfrac{R_{f2}}{R4} \\ \\ C = \dfrac{R_{f1}}{R1} \end{cases} \tag{4.18}$$

4.3.2 Law of exponential decay

As previously said, after analyzing and understanding the behavior of the generic linear differential equation there was an effort to find a real world problem where the knowledge of the generic equation could be put to work and it could be solved. Therefore the law of exponential decay seemed fitting for this application, as it describes a real world problem that is present in our daily lives as well as the equation is similar to the studied in the previous section, making the proposed circuit also a simple adaptation of the one presented in 4.3.1.

The law of exponential decay, which can be described by the expression 4.19, which is a general linear differential equation used to describe the rate at which the object of the study is decreasing, stating that this rate of change in a variable is proportional to the negative value of itself. It has been used in the field of physics to study the rate at which a radioactive body loses it effect as well as how a heated body loses it's temperature over time eventually reaching the same temperature as it's environment. It has also been used to study the change in a population over time, accounting for factors such as birth rates, deaths and migration [37].

$$N' = -\lambda N + k \tag{4.19}$$

From analyzing the expression 4.19 and comparing it to the 4.15 it is possible to establish a connection between the exponential decay constant, represented by λ, and the coefficient B. However this exponential decay equation adds the option to add a constant, for example, when solving the change in the effect of a radioactive body, this constant could represent outside factors that would have an implication on the change of this effect, such as any outside radiation or impurities in the sample. Which would mean that this constant could be represented by the coefficient C from the general equation, which would result in the following expression 4.20.

$$\begin{cases} \lambda = B = \dfrac{R_{f1}}{R2} * \dfrac{R_{f1}}{R4} \\ \\ k = C = \dfrac{R_{f1}}{R1} \end{cases} \tag{4.20}$$

4.3.3 Making it configurable

Now that the behavior of the circuit has been analyzed and it is understood how the equation translates itself to the circuit, with all the relations to each component established, as well as the connections made so that this equation has a meaningful use to the real world. This section will focus on how this circuit can be used to solve multiple linear differential equations of the same nature with varying coefficients.

As seen in equation 4.18, the resistors that impact the values of the coefficients, can be replaced with a parallel of switchable resistors. That would allow for the value of the equivalent resistor to be tuned to achieve the desired value of the corresponding coefficient, this is possible because by doing the parallel of two resistors it is possible to achieve any desired value as long as it is lower than the previous equivalent value.

Therefore three resistors from the circuit seen in Figure 4.8 that have an impact on the coefficients, which are R_1, R_2 and R_{f2}, were chosen to be replaced by the formerly mentioned parallel of switchable resistors that can be seen in Figure 4.9.

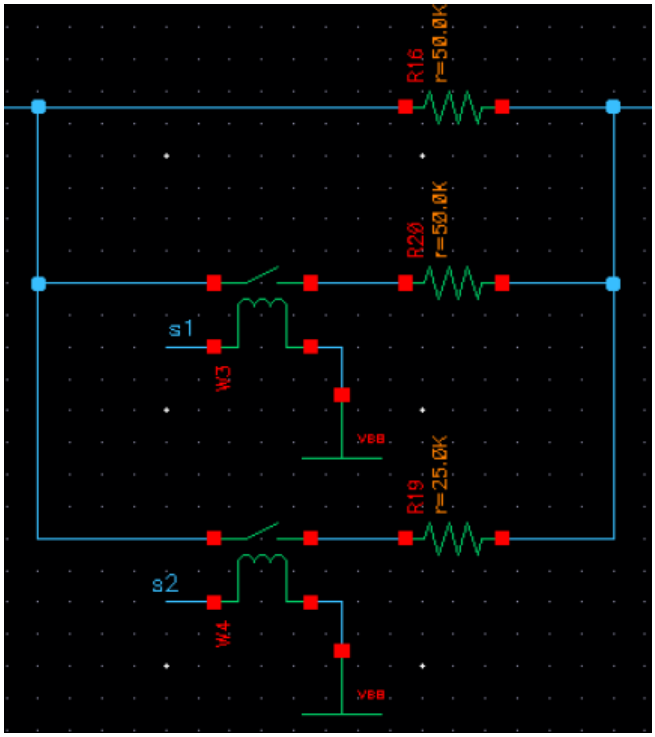


Figure 4.9: Design used for the switchable resistors

After designing this parallel block there was the need to design something that could control each of the switches, so that the values of the resistors could be tuned to the desired value. With this in mind an ADC is going to be used, so that by applying a fixed DC voltage the corresponding bits would turn to 1, making it so that, with some additional logic the equivalent switch would be closed, which resulted in the circuit seen in Figure 4.10.

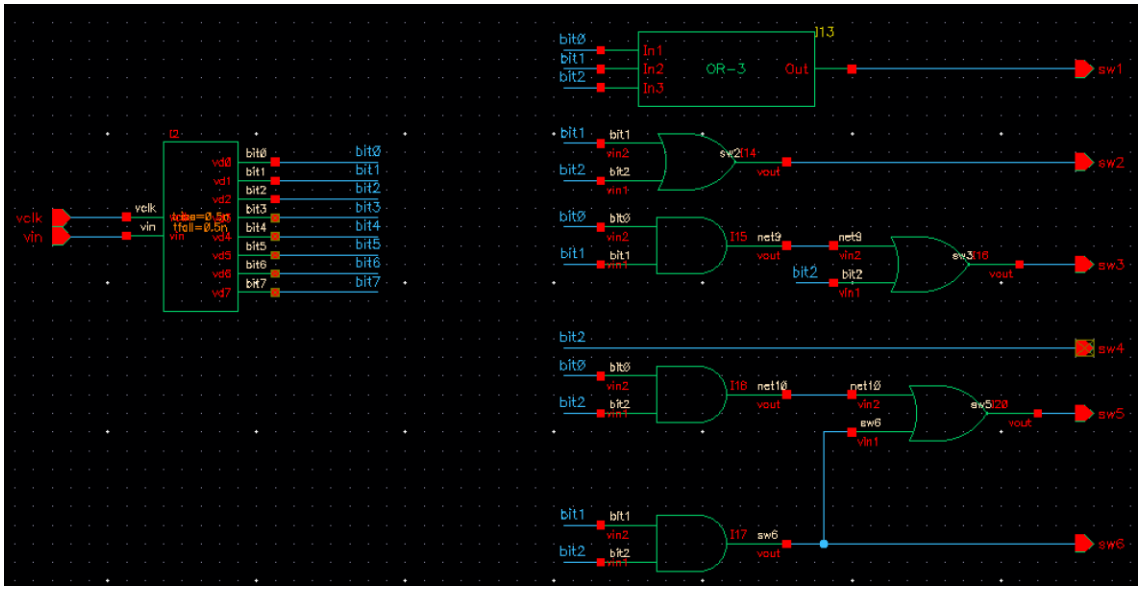


Figure 4.10: Circuit designed to control the switches

It was intended that each time the DC voltage was incremented in order to activate the next switch, all of the previous switches would still remain activated so that the parallels could work as intended and keep reducing the equivalent resistor value. For example, if the input voltage of the ADC was the value corresponding to close the third switch, this meant that switch one and two would remain closed, therefore to activate switch one it was created a three input OR logic gate that would close the switch when either of 3 bits were active.

With the testing of the ADC, it was possible to conclude that when a bit is active it produces a DC voltage equal to the reference voltage of the converter, which was set as the same value as VDD, 1.2V. With this in mind, the remaining thing to do was to correctly fill the parameters of the switches so that when the DC voltage came from the control block, these would properly close, and as seen in Figure 4.11 the open and close voltages were, respectively, 10mV below and above the 1.2V value.

Open voltage	1.19 V
Closed voltage	1.21 V

Figure 4.11: Parameters that allow for the proper function of the switch

Finally after analyzing what the behavior of the circuit in response to solving the equation should be, given their expressions, and how it was made possible to make this circuit programmable to be able to accept multiple linear differential equations of the same nature. In the next chapter, after verifying that the amplifier is working as intended in the integrator configuration by running the mentioned tests. Simulations from a first attempt at solving an equation with fixed coefficients will be shown, as well as in a later stage the results of the simulations done when changing the coefficients through the control block that was designed. At the same time these will be compared to a result given by an online differential equation solver.

5

Simulation results and validation

In this chapter, the simulations that were done to validate the results will be presented as well as the conclusions that can be taken from these. Firstly the simulations that were done to validate the theoretical results of the characteristics of the amplifier will be presented and analyzed. And lastly, as was explained in the previous chapter, there was a specific approach to this thesis, first a brief analysis of the integrator montage of an opamp, next a simple linear differential equation is to be solved and analyzed, and finally the results of making the circuit configurable will be shown.

When the results for the solving of the linear differential equation from the circuit done in Cadence's Virtuoso software, these simulations will also be compared to a real solution provided by an online differential equation solver. Therefore, for the next sections, each of them will present one of the formerly mentioned examples and the conclusions that were taken from them.

5.1 Operational amplifier

To begin this section, the schematics that were done using Cadence so that the characteristics of the circuit can be simulated will be presented, these were done with the reference design of Figure 4.1 which resulted in the schematic seen in Figure 5.1.

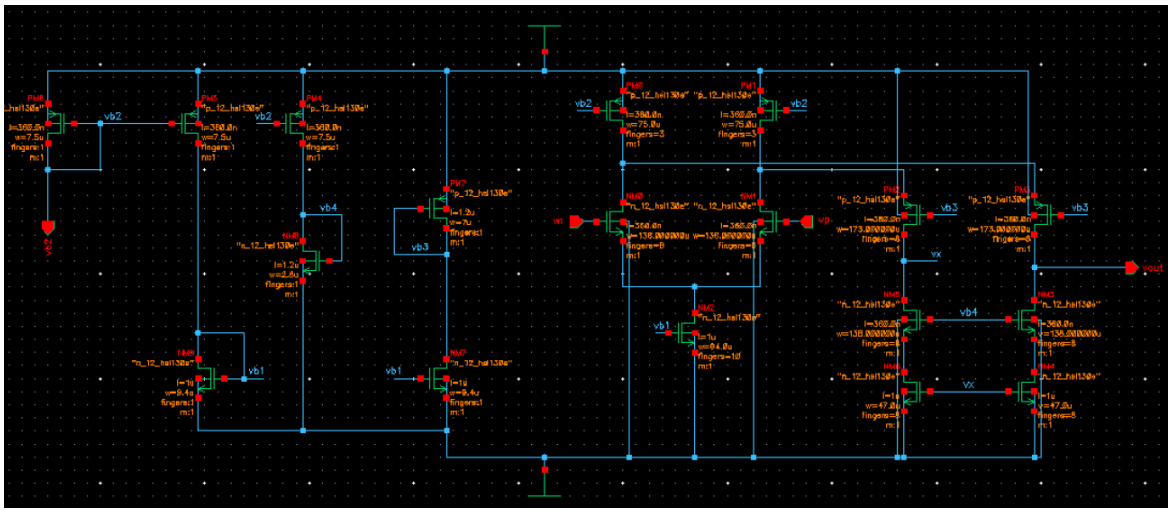


Figure 5.1: Schematic of a folded cascode in Cadence's software

After the parameters of each of the transistors that compose this architecture were introduced into the software, a schematic for the test-bench was designed to enable the simulations of the amplifier, which resulted in Figure 5.2. Which required that the components of the test-bench circuit were carefully filled out so that the amplifier was well supplied and it didn't have any stability issues, which were done by tuning the values of the input voltage supplies and output capacitor.

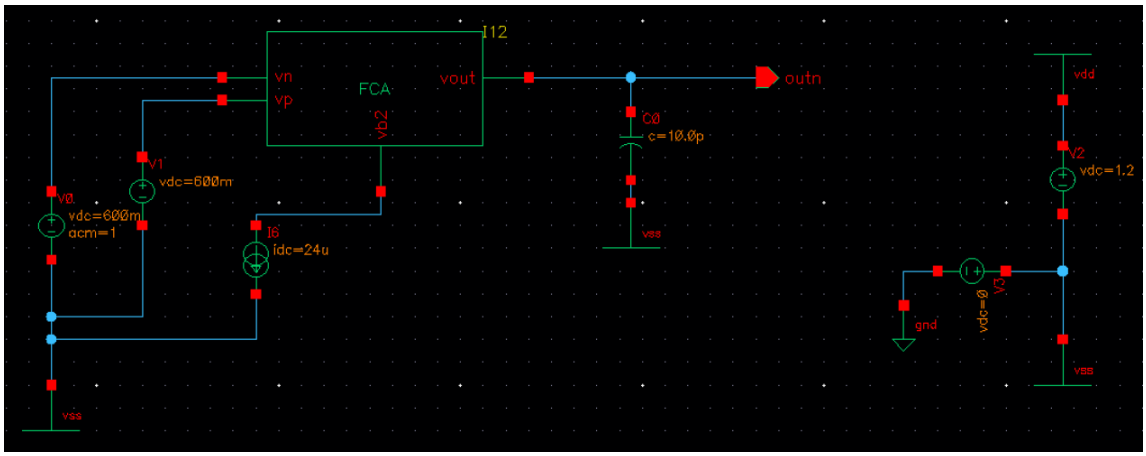


Figure 5.2: Schematic of the test-bench used for simualtions

Before getting into the specific analysis for each of the characteristics, an operating point simulation of the FCA was performed to guarantee that each transistor was operating in the moderate inversion zone. From this simulation it was possible to conclude that two pairs of transistors, M5/6 and M10/11, weren't in the desired operating zone, as the difference between their V_{dsat} and V_{ds} wasn't greater than the expected 80mV. Also the V_{dsat} from transistor M3/4 wasn't near the designed 100mV.

Therefore to adjust these values to their expected ones, there is the need to tune the width of the transistors that polarize them. Firstly the width of transistor MB6 was adjusted to 2.8 μm, from 1.2 μm, so that the V_{dsat} of M5/6 could be greater than their V_{ds} by the desired margin. Lastly by tuning the value of the width of the transistor MB5, which went from 12 μm to 7μm, it was possible to, not only get the V_{dsat} of M3/4 closer to it's designed value as well as the 80mV margin was achieved in the transistors M10/11.

Finally with the operating point of the circuit now working as intended it can be expected that in the AC simulations performed to the amplifier the results will be close to the ones gotten from the theoretical design of the circuit. These will be shown in the following sections.

5.1.1 DC gain

In Figure 5.3 it is possible to conclude that the DC gain obtained from the simulations of the opamp was pretty much the same as expected from the values obtained in the theoretical analysis. This was achieved by using a voltage source to introduce an alternate signal with 1V of amplitude and measuring the output signal in a logarithmic scale.

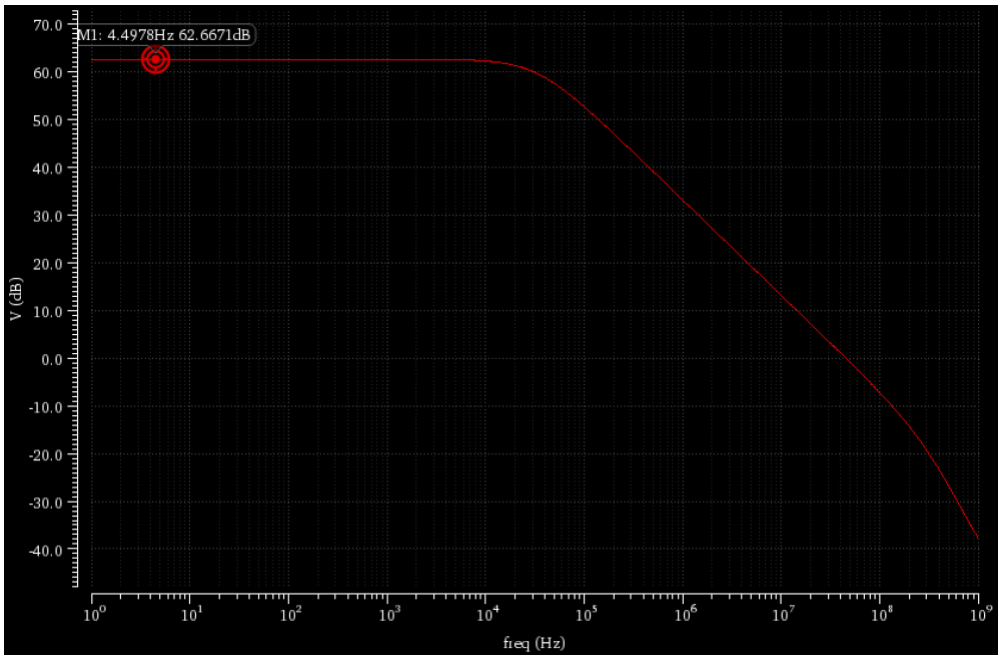


Figure 5.3: DC gain of the folded cascode amplifier

5.1.2 Gain Bandwidth Product

From Figure 5.4 it can be analyzed that the simulated GBW was slightly lower than the expected value by a margin of 10%, this was due to a small error in the V_{dsat} of the input transistors that was higher than the designed value.

This value was retrieved from the same simulation done in the previous section where it corresponds to the frequency where the output signal crosses 0dB.

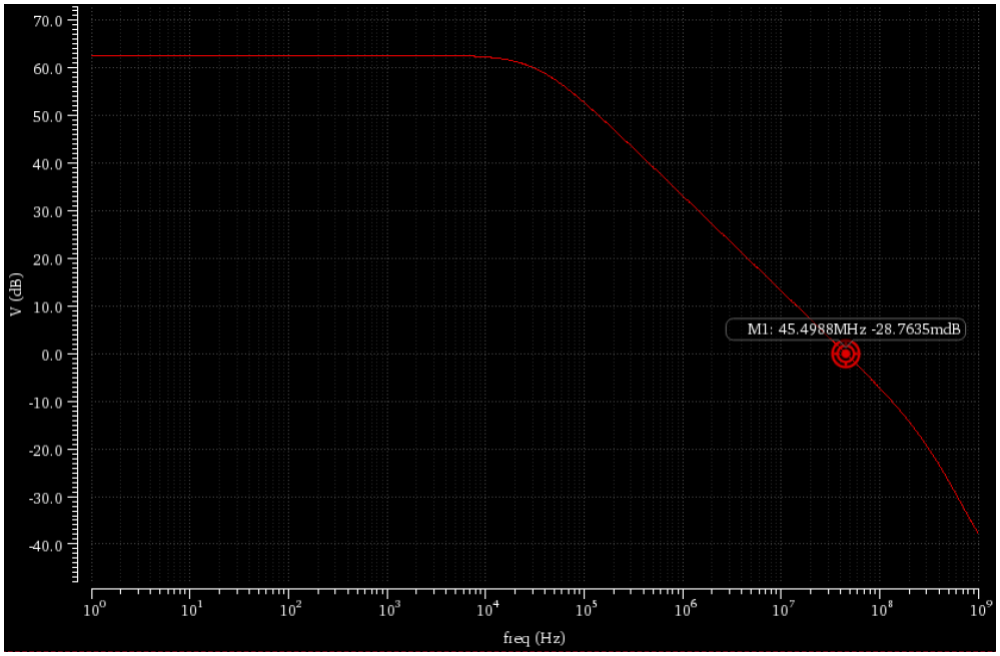


Figure 5.4: GBW of the folded cascode amplifier

5.2 Integrator simulations

To analyze the constraints of this circuit when dealing with a real amplifier two different tests were done, first an example of the integration of a sine wave and secondly the operation was performed on a square wave signal.

However, before starting with the simulations to validate that the integrator is working as intended, firstly the parameters of it's components must be defined. As explained in 4.2 a resistor was added to solve the reliability issues of using a real opamp by lowering it's low frequency gain. Due to this, for the circuit to work as an integrator the frequency of the input signal must be greater than the one given by the following expression 5.1, which relates this frequency to the product of both of the feedback components, the resistor and the capacitor.

$$f_1 = \frac{1}{2 * \pi * R_f * C_f} \tag{5.1}$$

In spite of that, the signal's frequency should also be lower that the GBW of the integrator, which is the frequency that relates to the feedback capacitor and input resistor, as seen in 5.2.

$$f_{0dB} = \frac{1}{2 * \pi * R_{in} * C_f} \tag{5.2}$$

With this is mind, the values that were chosen for the components to run these tests were extracted from [36], as seen in 5.3.

$$\begin{cases} R_{in} = 1k\Omega \\ R_f = 100k\Omega \\ C_f = 10nF \end{cases} \tag{5.3}$$

With these components and using the expressions 5.1 and 5.2, both of the frequencies that were previously mentioned values can be reached, which would be 159Hz and 15.9kHz, respectively.

Finally, before starting to run the simulations of the two chosen examples a frequency response analysis of the amplifier working as integrator was ran so that it's behavior could be analyzed based on the expected result. Which, from 5.5 can be taken that for f_1 and f_{0dB} both of the values were close to the desired, previously mentioned, values. Therefore the conditions to test this amplifier in it's integrator configuration were met and in the following two sections these simulations will be shown.

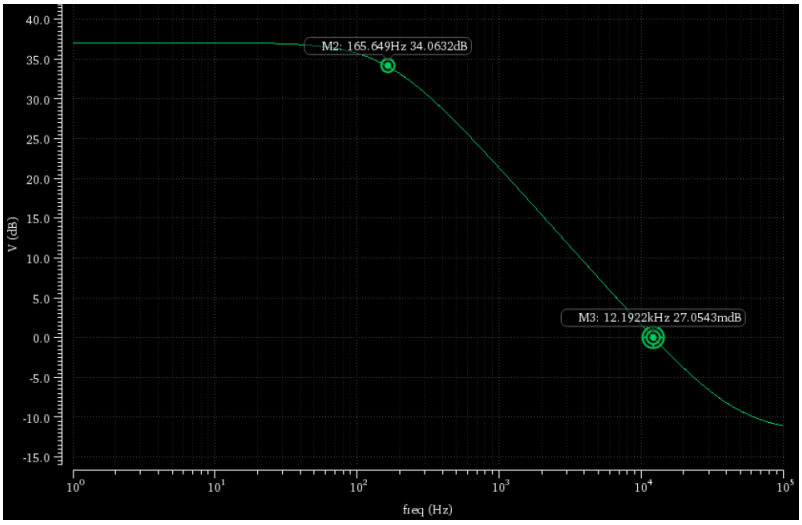


Figure 5.5: Frequency response of the integrator montage

5.2.1 Sine wave integration

Before the results of the simulation are presented, there is the need to first understand what these results are supposed to look like. Therefore a frequency of 5kHz and an

amplitude of 100mV will be considered for the input sine wave signal. With this in mind and following the equation 4.13, the output signal is expected to be described by the expression 5.4.

$$V_{out} = \frac{1}{2 * \pi * 5kHz * 1k\Omega * 10nF} cos(2 * \pi * 5kHz * t) \tag{5.4}$$

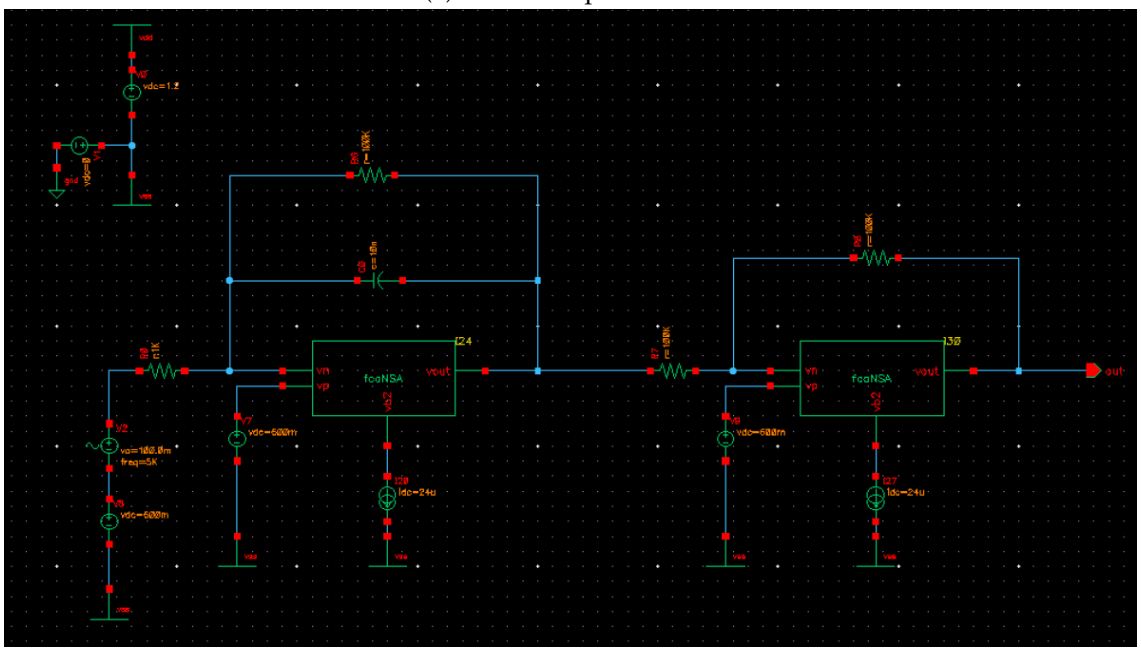
However since this integrator is set up in an inverting configuration this will mean that the output signal will have its phase inverted, which can be easily solved by adding an opamp as an inverter with unity gain. Also because this is a real amplifier which needs to be well polarized, the signal will be centered on it's common-mode voltage, 600mV. Which leads to the final expression 5.5 for the expected result of this test.

$$V_{out} = 0.6 + 0.318 * cos(2 * \pi * 5kHz * t) \tag{5.5}$$

In Figure 5.6a the parameters that were used to configure the vsine wave generator from Cadence can be observed, as for Figure 5.6b it shows the schematic that was implemented in order to simulate the integration of the sine wave.

Delay time	0 s
Offset voltage	
Amplitude	100.0m V
Initial phase for Sinusoid	
Frequency	5K Hz

(a) Sine wave parameters



(b) Schematic used for the simulation

Figure 5.6: Parameters and schematic used for the simulation of the sine wave

Lastly from Figure 5.7 it can be observed the result of the integration of this wave. From it is possible to conclude that even though the output wave is now a cosine as expected, because of it's 90 degree phase delay to the input signal, it does not have the expected amplitude, this being 250mV instead of the 318mV.

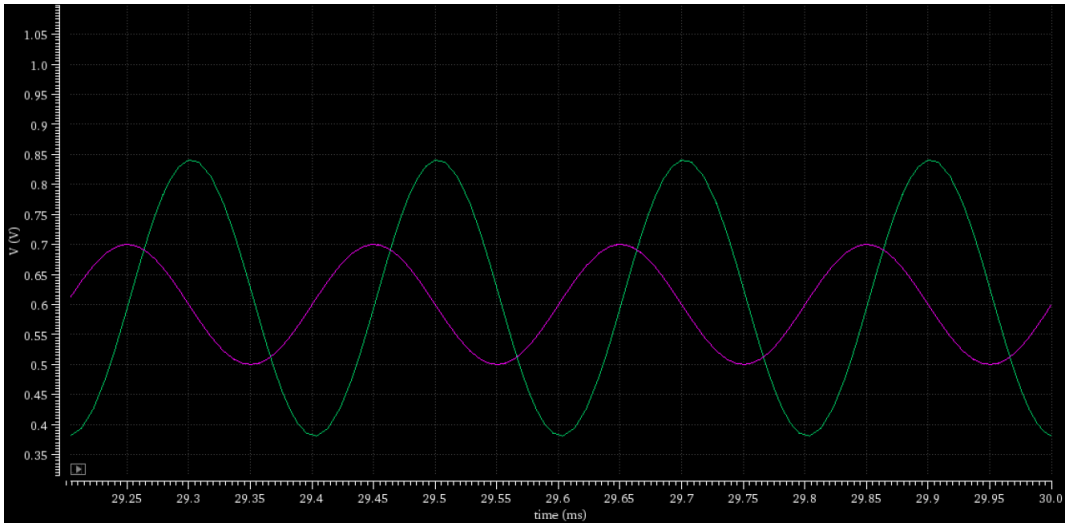


Figure 5.7: Result of the sine wave integration

5.2.2 Square wave integration

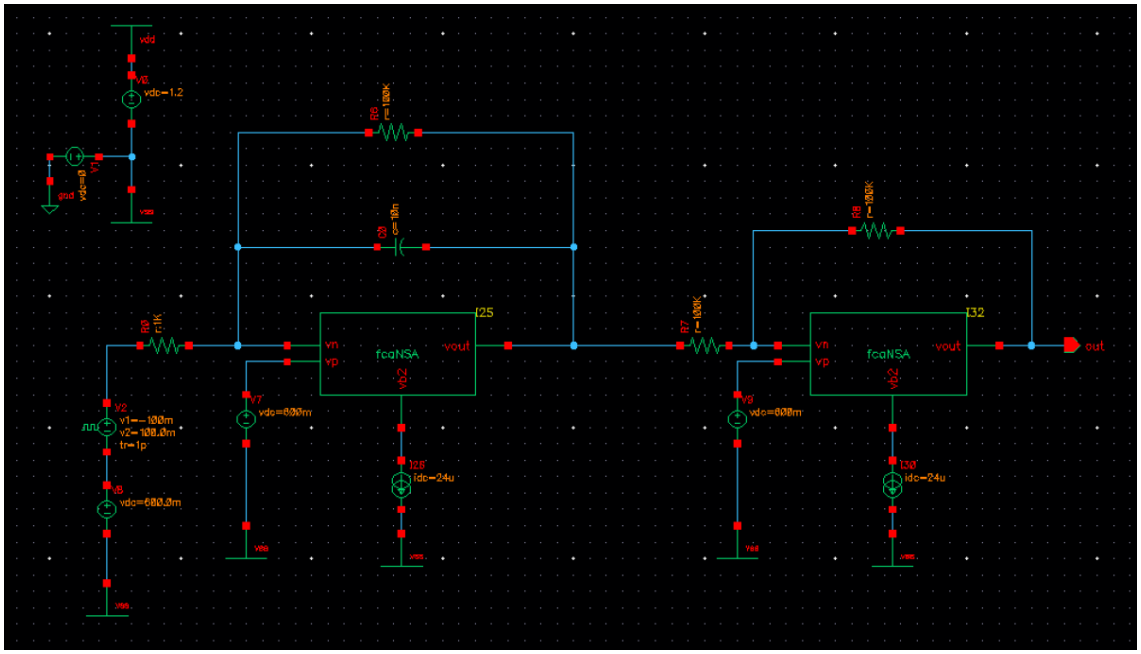
Likewise to the previous section, before starting to analyze the output of the simulations there is the need to first understand what to expect out of those results. For that, using the expression 4.14, where the input signal is going to be a square wave with 100mV of amplitude, 200 μs of period with no delay and negligible transition times, which would result in the equation found in 5.6.

$$\begin{cases} V_{in} = \pm 100mV \\ \\ V_{out} = \dfrac{1}{1k\Omega * 1nF}[50\mu s - 0]0.1 = \pm 250mV \end{cases} \tag{5.6}$$

From this expression it is possible to conclude that the output signal should have 2.5 times the amplitude of the input, but it should take the form of a triangular wave, since this is the result of integrating a square wave. An amplifier set up as inverter with unity gain was also necessary to get the output signal in the same phase as the input, as well as the output will be centered on the common mode voltage of 600mV.

Voltage 1	-100m V
Voltage 2	100.0m V
Period	200u s
Delay time	0 s
Rise time	1p s
Fall time	1p s
Pulse width	100u s

(a) Parameters of the pulse wave



(b) Schematic used to simulate the integration of a square wave

Figure 5.8: Parameters used for the simulation of the square wave and it's result

The parameters of the input square wave, that were previously mentioned, needed to configure the vpulse supply, can be observed in Figure 5.8a. As for the schematic used to simulate the integration of a square wave in can be seen in Figure 5.8

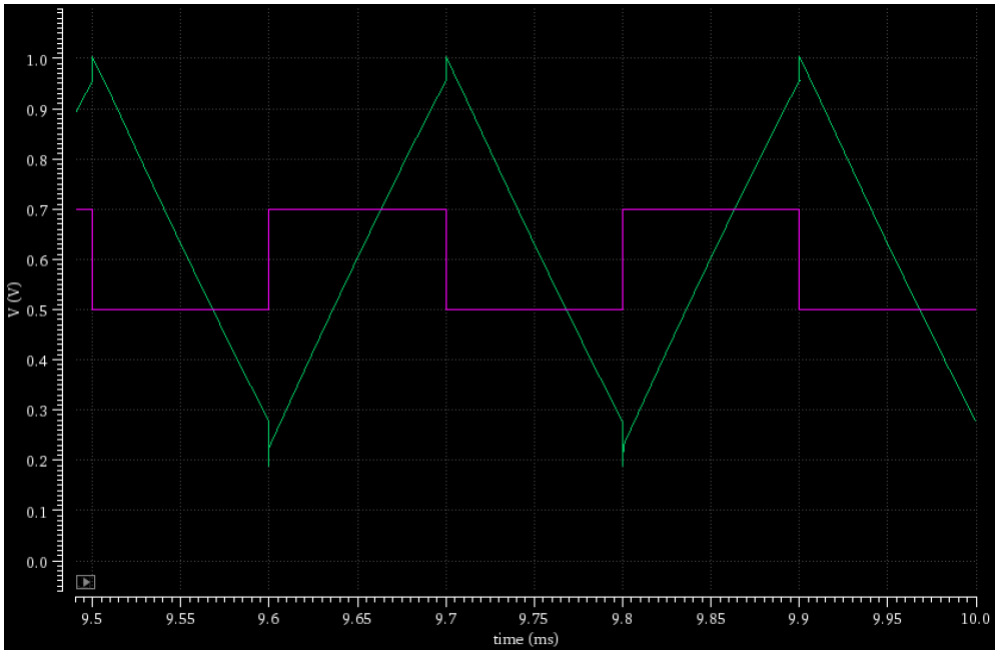


Figure 5.9: Simulation results from the integration of a square wave

Lastly, Figure 5.9 shows the result of the simulation of this input wave, where it can be analyzed that the output wave, as expected, is a triangular wave, although the amplitude of the result was over the expected value, this being 250mV while the result was 350mV.

5.3 Linear differential equation

Finally it comes the time to actually solve a differential equation using an analog circuit, for that, in the next sections, in a first effort a linear differential equation will be presented, after that the work will progress in order to make the circuit as configurable as possible, as explained in the previous chapter.

With this in mind, an example of a linear differential equation was extracted from [35], from where the coefficients to the equation, analyzed in the previous chapter, were taken, from here the circuit that describes it will be shown and then the simulations to validate the simulated outcome. Finally, in the last section, a few more examples will be solved with different coefficients relying on the control circuit that was shown as well as the simulations that prove it's reliable function.

5.3.1 No configuration

To begin these final sections where the solving of a differential equation is done, firstly there is the need to define the coefficients that make up the equation 4.15. To make this possible and as previously mentioned, an example of a linear differential equation was taken from [35], expression 5.7, which was then used as reference to understand how these values will define the components of the circuit as well as what the outcome should be.

$$y'(t) = -(5y(t) + 2) \tag{5.7}$$

By analyzing this equation and comparing it to 4.15, it is possible to extract the following coefficients, seen in 5.8.

$$\begin{cases} A = 1 \\ B = 5 \\ C = 2 \end{cases} \tag{5.8}$$

From here and bearing in mind the relation that was previously established between these coefficients and the components of the circuit in 4.18, it is now possible to define which values will be used for the components of the circuit, the resistors, as well as the DC voltage that will be used to trigger a step response of the circuit, as shown in 5.9

$$\begin{cases} R_1 = 50k\Omega \\ R_2 = 20k\Omega \\ R_{f1} = 100k\Omega \\ V_1 = 100mV \end{cases} \tag{5.9}$$

With these components now defined and using 100kΩ as the value for both of the resistors of the unity gain inverter, there is a last step to be done before the simulations can be done. The components of the integrator must be defined, for this the values used in a first attempt were same as values used in tests done to the amplifier in it's integrator configuration, which proved successful when the simulations were performed, which meant no tuning of these values was performed. Which in turn resulted in the circuit shown in Figure 5.10.

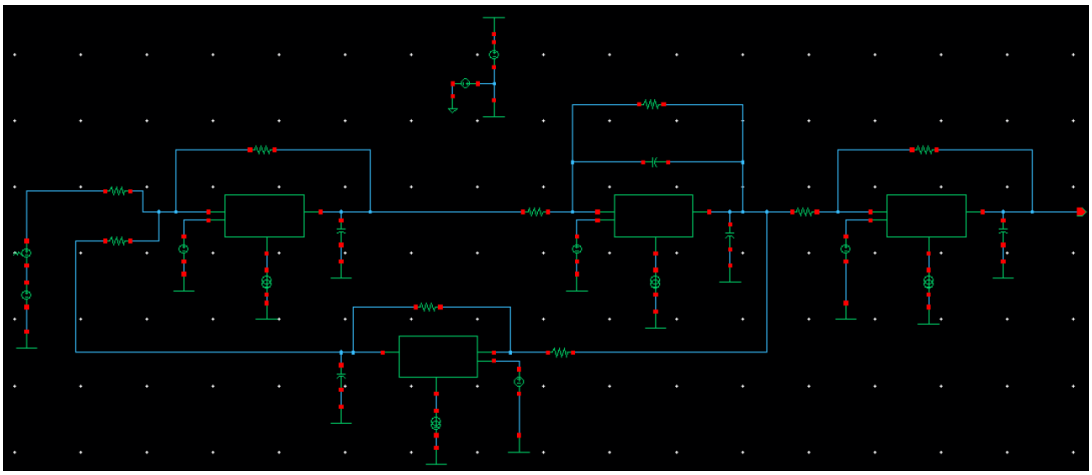
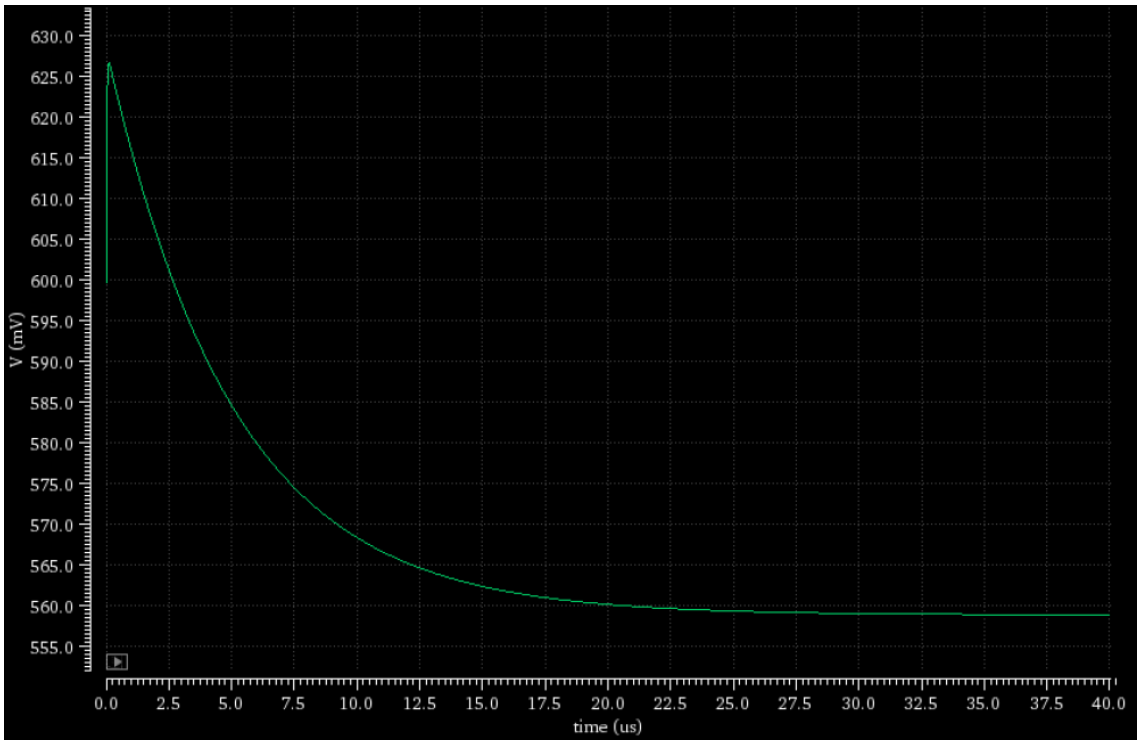


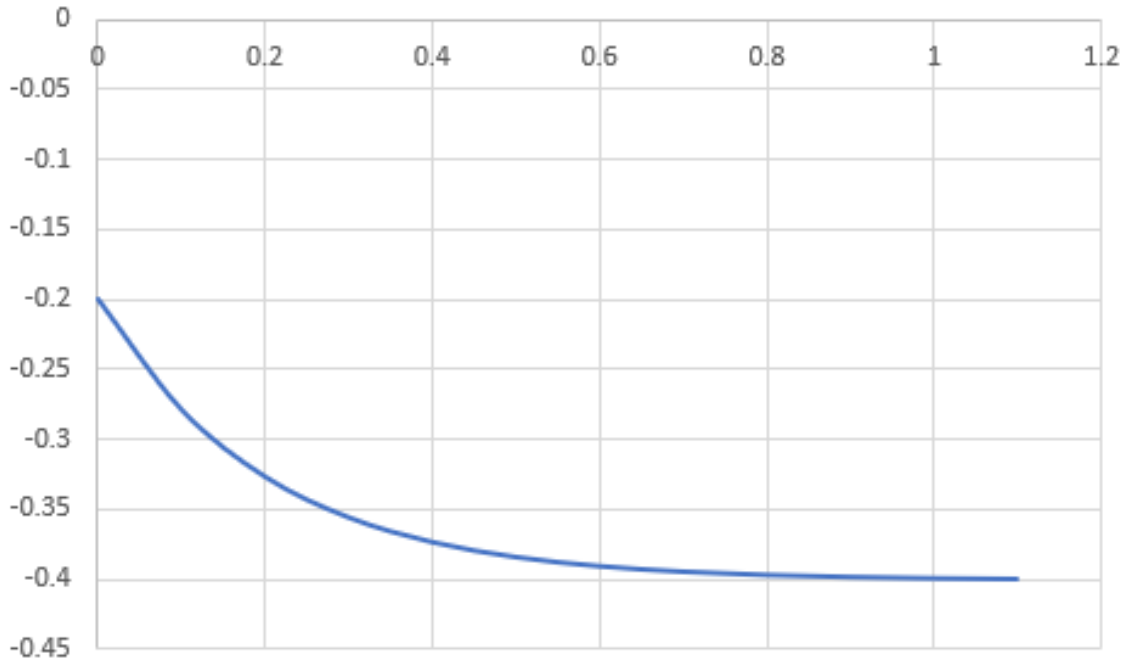
Figure 5.10: Circuit that defines the differential equation

Before simulations were performed, there is the need to understand how the circuit will start, for this and because, unlike other Spice simulators, Cadence's Virtuoso software doesn't have a start DC supplies from zero which was needed for the input voltage step signal, a pwl voltage supply was used to create this step signal so that the step response to this impulse can be seen at the output which would be the solution to the equation.

As previously mentioned, the solution provided by the analog circuit will be compared to a solution computed by an online differential equation solver. Therefore as can be seen in Figure 5.11a, which represents the solution gotten from the simulation of the circuit, where the input step signal was equal to negative 100mV, this being because the constant in the expression is also negative in value. It can be seen that if it were possible to subtract the common-mode voltage, it comes to the same output as the latter, Figure 5.11b, which is a good result for the analog circuit.



(a) Circuit's solution



(b) Online solver solution

Figure 5.11: Comparison between the circuit's solution and differential equation solver solution

Lastly, before moving to last section of this chapter, the time that took each of these simulations to happen and the power consumption while doing so must be compared, since these are the strengths of an analog computer when tasked to run a repeatable algorithm. The time that the digital computer took to solve this equation was measured using a Matlab/Octave script, also while using a software tool to measure the power consumption of this computer. For the power consumption of the analog computer only an estimate can be had using the power consumption estimated by the Mathcad software at the time of the theoretical design of the opamp.

	Time	Power Consumption
Analog computer	$40\mu s$	$40mW$
Digital computer	$0.33s$	$72.6W$

Table 5.1: Comparison of solution time and power consumption

From table 5.1 it is possible to conclude that as expected the analog computer beats the digital computer both in time and power consumption.

5.3.2 With configuration

Finally in this last section, the simulations that were done to the programmable circuit are going to be shown to prove that by controlling the switches it is possible to change the values of the coefficients. By changing the value of the decay rate or the constant through the values of the components that are related to them respectively, it is possible to analyze different behaviors for a problem.

As an example, by changing the value of the resistor R_2 or R_4 it is possible to change the rate at which a radioactive body loses it's effect. As for the value of the resistor R_1, if this is varied it can mean that there was a change on the outside factors that can have an impact on the change of this effect as previously mentioned.

With this in mind, on a first stage there was the need to define the order as to which resistors would be affected. Settling on that firstly the values of resistor R_2 would be lowered so that the decay rate could be changed, next the switches that lower the value of the constant resistor R_1 would be activated, and lastly the decay rate would be increased once again by changing the value of the equivalent resistor R_4.

Therefore three tests, all with different objectives which will be explained as they are presented, where both the decay rate and the constant will have different values in order to validate that this system can solve multiple exponential decay equations by changing the applied control voltage. In the following tables 5.2, 5.3 and 5.4, the equivalent resistor value change is shown in the relation to the change in the control voltage.

$V_{control}\ (mV)$	$Resistor$
5	$R_2 = 50k\Omega$
10	$R_2 = 25k\Omega$
15	$R_1 = 25k\Omega$
20	$R_1 = 12.5k\Omega$
25	$R_4 = 50k\Omega$
30	$R_4 = 25k\Omega$

Table 5.2: Resistor values for test 1

$V_{control}\ (mV)$	$Resistor$
5	$R_2 = 66.7k\Omega$
10	$R_2 = 50k\Omega$
15	$R_1 = 40k\Omega$
20	$R_1 = 33.3k\Omega$
25	$R_4 = 80k\Omega$
30	$R_4 = 60k\Omega$

Table 5.3: Resistor values for test 2

$V_{control}\ (mV)$	$Resistor$
5	$R_2 = 12.5k\Omega$
10	$R_2 = 8.33k\Omega$
15	$R_1 = 2.5k\Omega$
20	$R_1 = 2k\Omega$
25	$R_4 = 80k\Omega$
30	$R_4 = 50k\Omega$

Table 5.4: Resistor values for the last test

There is one more final step before the simulations are done and the results are presented, which is to translate these changes in resistor values to the equation of the law of exponential decay, so that when the simulations are done the final result can be compared to result given by the online differential equation solver. Consequently the following tables, 5.5, 5.6 and 5.7, show the relation between the control voltage and the equivalent equation.

$V_{control}\ (mV)$	$Expression$
0	$y' = -y - 2$
5	$y' = -2y - 2$
10	$y' = -4y - 2$
15	$y' = -4y - 4$
20	$y' = -4y - 8$
25	$y' = -8y - 8$
30	$y' = -16y - 8$

Table 5.5: Expressions for test 1

$V_{control}\ (mV)$	$Expression$
0	$y' = -y - 2$
5	$y' = -3/2y - 2$
10	$y' = -2y - 2$
15	$y' = -2y - 5/2$
20	$y' = -2y - 3$
25	$y' = -5/2y - 3$
30	$y' = -5y - 3$

Table 5.6: Expressions for test 2

$V_{control}\ (mV)$	$Expression$
0	$y' = -5y - 20$
5	$y' = -8y - 20$
10	$y' = -12y - 20$
15	$y' = -12y - 40$
20	$y' = -12y - 50$
25	$y' = -15y - 50$
30	$y' = -24y - 50$

Table 5.7: Expressions of the last test

As per the previous section, here the simulations were also done with a pwl voltage source to obtain a step response from the system, also with a negative 100mV value because of the previously explained reason. So in order to finish off this section, the results for each of the mentioned tests will be shown in their respective order.

In the conditions of the first test, it becomes clear that the final solution depends on the ratio between both of the coefficients. This can be seen in Figure 5.12 and in the table 5.8 where when, for example, the decay rate is double that of the constant the final response will always be around 550mV, although the speed at which the system gets there depends only on the decay rate as expected.

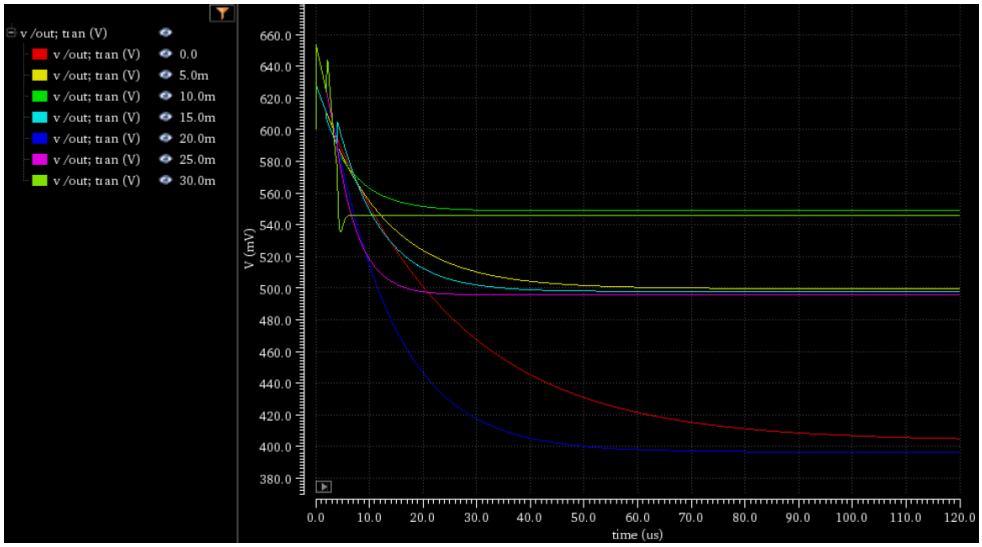


Figure 5.12: Simulation results in the first condition

In table 5.8 it can also be observed that the system, when compared to the online differential equation solver, solves the multiple equations with really good precision, this being in the worst case scenario, for these conditions, equal to 1%. However the results from the online solver were adjusted so that a comparison could be made, this means that the results extracted were adjusted to simulate the common-mode voltage of the amplifier of 600mV.

Expression	*Expected (mV)*	*Simulated (mV)*	*Error %*
$y' = -y - 2$	400	404	1
$y' = -2y - 2$	500	499	0.2
$y' = -4y - 2$	550	548	0.36
$y' = -4y - 4$	500	497	0.6
$y' = -4y - 8$	400	396	1
$y' = -8y - 8$	500	495	1
$y' = -16y - 8$	550	545	0.9

Table 5.8: Simulations results compared to ideal results for the first test

For the second test, the objective was to test a wider range of ratios between the two mentioned coefficients in order to avoid getting multiple equal results, which can be seen in Figure 5.13 and table 5.9. Not only were all of the final solutions different from each other, but also the system again managed to solve them with really good precision, where in the worst case the error of the circuit versus the online solver was only one percent, even having two results with no error.

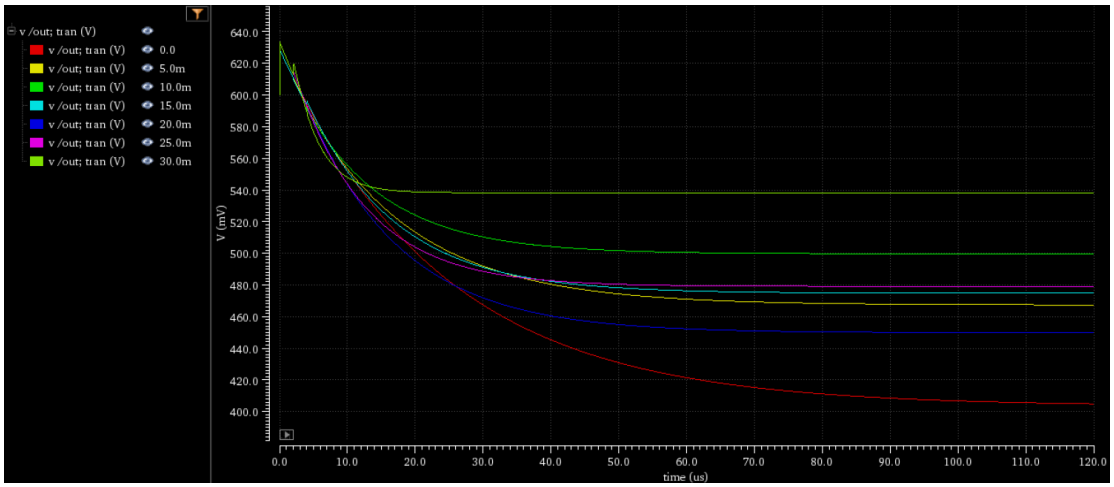


Figure 5.13: Simulation results in the second condition

Expression	*Expected (mV)*	*Simulated (mV)*	*Error %*
$y' = -y - 2$	400	404	1
$y' = -3/2y - 2$	467	467	0
$y' = -2y - 2$	500	499	0.2
$y' = -2y - 5/2$	475	474	0.2
$y' = -2y - 3$	450	450	0
$y' = -5/2y - 3$	480	479	0.2
$y' = -5y - 3$	540	538	0.4

Table 5.9: Simulations results compared to ideal results for the second test

Finally for the last test, the objective was not only to keep the results as different from each other as possible, by having the most ratios between the decay rate and the constant, but also to get higher values of the decay rate as well as the constant. Which, as can be seen in Figure 5.14, both of these objectives were achieved and the system was able to solve all of the equations.

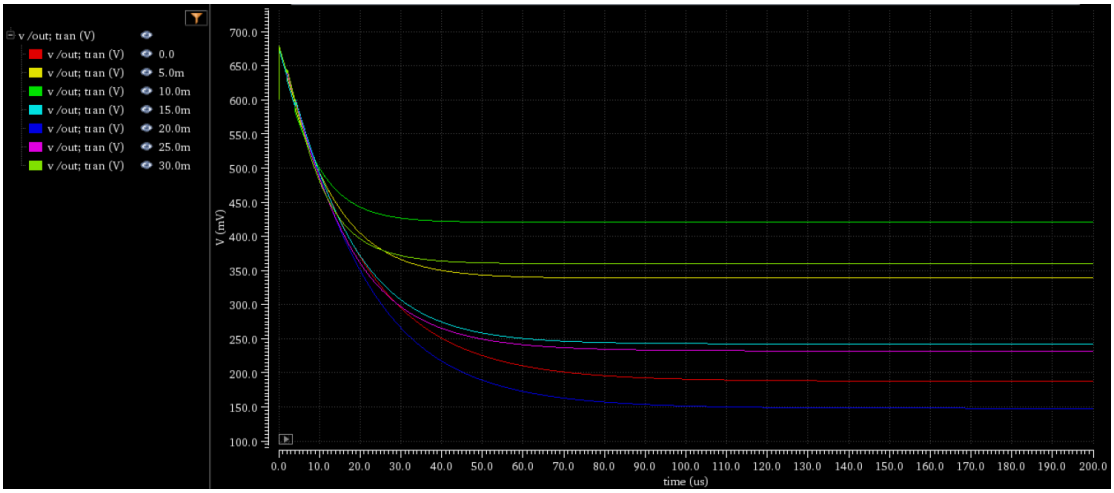


Figure 5.14: Simulation results in the last condition

From table 5.10 it is possible to conclude that the system, when compared to the online differential equation solver, starts to get bigger errors as the values of both the decay rate and the constant increase, starting of with acceptable precision but almost reaching 20% of error in one of the solutions. This worst case being when both the decay rate and the constant have not only high values but also the biggest ratio between them.

Expression	*Expected (mV)*	*Simulated (mV)*	*Error %*
$y' = -5y - 20$	200	187.5	6.25
$y' = -8y - 20$	350	338.6	3.26
$y' = -12y - 20$	433.3	420.6	2.93
$y' = -12y - 40$	266.7	242.3	9.15
$y' = -12y - 50$	183.4	147.9	19.36
$y' = -15y - 50$	266.7	231.9	13.1
$y' = -24y - 50$	391.7	359.6	8.2

Table 5.10: Simulations results compared to ideal results for the third test

From these three last simulations the only notable change was the time it took the analog computer to reach a solution for each equation, where times ranged from 30μs to around 130 μs. While the time has increased for some of the equations it was still way below the time that the digital computer took to solve the problems. As for the power consumption of both of these devices it was estimated that it remained unchanged from the results presented in 5.1.

6

Conclusions and Future work

In this final chapter some conclusions will be taken from all of the work that was done to complete this thesis, explaining what was learnt and describing what were the problems and shortcomings that were encountered because there were a couple. Finally in the last section the future work that could be done with what was learnt from this thesis.

6.1 Conclusions

Even though analog computers have existed for a long period of time, they have now began to resurface after long years in the dark because of the invention and development of digital computers. These started to be used again due to the fact that CMOS technologies became more mainstream in the past few years, because of that it was noticed that an analog computer could be extremely efficient and fast at solving differential equations, which are the base of most of the problems being solved today in various areas of science and engineering.

Therefore, in this dissertation it was presented a circuit that described a simple linear first order differential equation that represents a variation of the law of exponential decay. To said circuit was then added a logic circuit and some switched resistors in order for it to be able to solve the same equation but with varying coefficients.

However before this circuit could be built, there was the need to first design it's core, the opamp. For this a FCA montage was analyzed, designed and then simulated to validate it's design, where it's possible to draw a first conclusion, that the values targeted for the main characteristics of this amplifier were met with success.

After having a working amplifier it was time to move to understanding the integrator montage of the FCA, this is where the first problems were met, even though it seemed that the integrator was well designed to be able to integrate the test signal. When the simulations were done, although the output signals had the expected form, the amplitudes didn't quite match with the theoretical values. However these results played no effect when it came to the part of solving the chosen differential equation, because in the first two tests the accuracy of the solver was within 1% of the theoretical result.

Leaving only the last test that was done where the solver was expected to start to show some weaknesses, because this FCA doesn't do well with low impedances at the input, which were required to be able to reach higher coefficients. This lead to higher inaccuracies in this last test, reaching as high as almost 20% in the worst case that was solved, which leads to the conclusion that in order for this error to be inferior to 10% the input resistor R_1 should have a value lower that 5kΩ.

To finalize, since analog computers are mostly used for routine work, which make time and power efficiency the most important conclusion from this work, where in both of these characteristics the analog computer reported better results by an order of magnitude versus the digital computer. The former reaching solution times of 30μs where the latter took 0.3s, while in power consumption the values reached were estimated to be 40mW and 72W, respectively.

6.2 Future work

This dissertation due to the fact that it showed some promising results it opens the door for a few future work scenarios. The first one being that the proposed circuit could be built in integrated circuit so that the results of this physical prototype could be compared to the simulations.

The other proposed future work relate to the optimizing of this circuit and building upon it, firstly a new architecture for the amplifier could be studied to see if it has the same constraints of the FCA in terms of limiting the values of the coefficients. This could also be achieved by changing the design of the integrator, where a more complex integrator topology could used to see if that would help in improving the coefficient constraint.

And lastly the circuit could be adapted to solve similar linear first order equation or taken a step further in the hopes of solving non-linear higher order equations, for example by designing a multiplier and introducing it to this circuit.

Bibliography

[1] J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/master/template.pdf (cit. on p. ii).

[2] Y. Tsividis. "Not your Father's analog computer". In: *IEEE Spectrum* 55.2 (2018), pp. 38–43. DOI: 10.1109/MSPEC.2018.8278135 (cit. on pp. 1, 2).

[3] N. Guo et al. "Energy-Efficient Hybrid Analog/Digital Approximate Computation in Continuous Time". In: *IEEE Journal of Solid-State Circuits* 51.7 (2016), pp. 1514–1524. DOI: 10.1109/JSSC.2016.2543729 (cit. on pp. 1, 13, 15).

[4] K. H. Lundberg. "The history of analog computing: introduction to the special section". In: *IEEE Control Systems Magazine* 25.3 (2005), pp. 22–25. DOI: 10.1109/MCS.2005.1432595 (cit. on pp. 1, 3).

[5] A. Taflove and S. Hagness. *Computational Electrodynamics: The Finite-Difference Time-Domain Method, 3rd edition*. Vol. 2062. 2005-02. ISBN: 978-1-58053-832-9. DOI: 10.1016/B978-012170960-0/50046-3 (cit. on p. 1).

[6] K. Yee. "Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media". In: *IEEE Transactions on Antennas and Propagation* 14.3 (1966), pp. 302–307. DOI: 10.1109/TAP.1966.1138693 (cit. on p. 1).

[7] G. Mur. "Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations". In: *IEEE Transactions on Electromagnetic Compatibility* EMC-23.4 (1981), pp. 377–382. DOI: 10.1109/TEMC.1981.303970 (cit. on p. 1).

[8] W. F Ames. *Nonlinear Partial Differential Equations in Engineering*. ISSN. Elsevier Science, 1965. ISBN: 9780080955247. URL: https://books.google.pt/books?id=xwaZP3fxAzIC (cit. on p. 1).

[9] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*. eng. 2nd ed. Berlin: Springer, 2006. ISBN: 978-3-540-30663-4. URL: https://archive-ouverte.unige.ch/unige:12343 (cit. on p. 1).

[10] J. M. Sanz-Serna. "Runge-kutta schemes for Hamiltonian systems". In: *BIT* 28.4 (1988-12), pp. 877–883. DOI: 10.1007/bf01954907. URL: https://doi.org/10.1007/bf01954907 (cit. on p. 1).

[11] N. Udayanga et al. "Continuous-Time Algorithms for Solving Maxwell's Equations Using Analog Circuits". In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 66.10 (2019), pp. 3941–3954. DOI: 10.1109/TCSI.2019.2915200 (cit. on pp. 1, 10).

[12] DataGrapha. *Moore's Law graphed vs real CPUs & GPUs 1965 - 2019*. URL: https://www.youtube.com/watch?v=7uvUiq_jTLM (cit. on p. 1).

[13] N. Loubet et al. "Stacked nanosheet gate-all-around transistor to enable scaling beyond FinFET". In: *2017 Symposium on VLSI Technology*. 2017, T230–T231. DOI: 10.23919/VLSIT.2017.7998183 (cit. on p. 1).

[14] H. J. Caulfield and S. Dolev. "Why future supercomputing requires optics". In: *Nature Photonics* 4.5 (2010-05), pp. 261–263. DOI: 10.1038/nphoton.2010.94. URL: https://doi.org/10.1038/nphoton.2010.94 (cit. on p. 1).

[15] S. Abdollahramezani, O. Hemmatyar, and A. Adibi. "Meta-optics for spatial optical analog computing". In: (2020). DOI: 10.48550/ARXIV.2007.14568. URL: https://arxiv.org/abs/2007.14568 (cit. on p. 1).

[16] M. A. Zidan et al. "A general memristor-based partial differential equation solver". In: *Nature Electronics* 1.7 (2018-07), pp. 411–420. DOI: 10.1038/s41928-018-0100-6. URL: https://doi.org/10.1038/s41928-018-0100-6 (cit. on p. 2).

[17] G. E. R. Cowan, R. C. Melville, and Y. P. Tsividis. "A VLSI analog computer/digital computer accelerator". In: *IEEE Journal of Solid-State Circuits* 41.1 (2006), pp. 42–53. DOI: 10.1109/JSSC.2005.858618 (cit. on p. 2).

[18] Y. Huang et al. "Analog Computing in a Modern Context: A Linear Algebra Accelerator Case Study". In: *IEEE Micro* 37.3 (2017), pp. 30–38. DOI: 10.1109/MM.2017.55 (cit. on p. 2).

[19] N. Guo et al. "Continuous-time hybrid computation with programmable nonlinearities". In: *ESSCIRC Conference 2015 - 41st European Solid-State Circuits Conference (ESSCIRC)*. 2015, pp. 279–282. DOI: 10.1109/ESSCIRC.2015.7313881 (cit. on p. 2).

[20] A. C. King, J. Billingham, and S. R. Otto. *Differential equations*. Cambridge, England: Cambridge University Press, 2003-05 (cit. on p. 5).

[21] G. Han and E. Sanchez-Sinencio. "CMOS transconductance multipliers: a tutorial". In: *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 45.12 (1998), pp. 1550–1563. DOI: 10.1109/82.746667 (cit. on p. 8).

[22] R. Baker. *CMOS: Circuit Design, Layout, and Simulation, Third Edition*. Vol. 18. 2010-02. ISBN: 978-0470881323. DOI: 10.1002/9780470891179 (cit. on pp. 8, 21).

[23] N. I. Khachab and M. Ismail. "A nonlinear CMOS analog cell for VLSI signal and information processing". In: *IEEE Journal of Solid-State Circuits* 26.11 (1991), pp. 1689–1699. DOI: 10.1109/4.98991 (cit. on pp. 8, 21).

[24] M. Ismail et al. "A configurable CMOS multiplier/divider for analog VLSI". In: *1993 IEEE International Symposium on Circuits and Systems*. 1993, pp. 1085–1088. DOI: 10.1109/ISCAS.1993.393923 (cit. on pp. 8, 21).

[25] H. Malavipathirana et al. "A Fast and Fully Parallel Analog CMOS Solver for Nonlinear PDEs". In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 68.8 (2021), pp. 3363–3376. DOI: 10.1109/TCSI.2021.3085214 (cit. on pp. 9, 21, 22).

[26] N. Udayanga et al. "A Radio Frequency Analog Computer for Computational Electromagnetics". In: *IEEE Journal of Solid-State Circuits* 56.2 (2021), pp. 440–454. DOI: 10.1109/JSSC.2020.3018432 (cit. on pp. 10, 16, 18).

[27] N. Udayanga et al. "Continuous-time Analog Computing Circuits for Solving The Electromagnetic Wave Equation". In: *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. 2018, pp. 1–5. DOI: 10.1109/ISCAS.2018.8351252 (cit. on p. 10).

[28] G. E. R. Cowan, R. C. Melville, and Y. P. Tsividis. "A VLSI analog computer/math co-processor for a digital computer". In: *ISSCC. 2005 IEEE International Digest of Technical Papers. Solid-State Circuits Conference, 2005.* 2005, pp. 82–586. DOI: 10.1109/ISSCC.2005.1493879 (cit. on p. 13).

[29] J. Liang et al. "An Offset-Cancelling Discrete-Time Analog Computer for Solving 1-D Wave Equations". In: *IEEE Journal of Solid-State Circuits* 56.9 (2021), pp. 2881–2894. DOI: 10.1109/JSSC.2021.3074003 (cit. on pp. 18, 20).

[30] A. Bressan et al., eds. *Nonlinear Conservation Laws and Applications*. Springer US, 2011. DOI: 10.1007/978-1-4419-9554-4. URL: https://doi.org/10.1007/978-1-4419-9554-4 (cit. on p. 20).

[31] R. W. MacCormack. "Numerical solution of the interaction of a shock wave with a laminar boundary layer". In: *Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics*. Ed. by M. Holt. Berlin, Heidelberg: Springer Berlin Heidelberg, 1971, pp. 151–163. ISBN: 978-3-540-36533-4 (cit. on p. 20).

[32] S. I. Hariharan and H. C. Lester. "Acoustic shocks in a variable area duct containing near sonic flows". In: *Journal of Computational Physics* 58.1 (1985), pp. 134–145. ISSN: 0021-9991. DOI: https://doi.org/10.1016/0021-9991(85)90161-5. URL: https://www.sciencedirect.com/science/article/pii/0021999185901615 (cit. on p. 20).

[33] D. Gottlieb, E. Turkel, and S. Abarbanel. "Analysis of the Error for Approximations to Systems of Hyperbolic Equations". In: *Journal of Computational Physics* 151.2 (1999), pp. 997–1007. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.1999.6196. URL: https://www.sciencedirect.com/science/article/pii/S002199919996196X (cit. on p. 20).

[34] Y. Tsividis and C. McAndrew. *Operation and Modeling of the MOS Transistor*. The Oxford Series in Electrical and Computer Engineering Series. Oxford University Press, 2011. ISBN: 9780195170153. URL: https://books.google.pt/books?id=oYmYPwAACAAJ (cit. on p. 21).

[35] B. Ndanusa et al. *Solving first Order Linear Differential Equation Using Operational Amplifier Approach*. Tech. rep. 1. 2017, pp. 2536–7323. URL: http://www.casirmediapublishing.com (cit. on pp. 25, 36, 51).

[36] ALLABOUTELECTRONICS. *Op-Amp Integrator (with Derivation and Solved Examples)*. URL: https://youtu.be/OPvs7A554Rw (cit. on pp. 33, 45).

[37] M. Vivas-Cortez et al. "Newton's law of cooling with generalized conformable derivatives". In: *Symmetry* 13.6 (2021-06). ISSN: 20738994. DOI: 10.3390/sym13061093 (cit. on p. 37).