UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ERIC ERICSON FABRIS

A Modular and Digitally Programmable Interface Based on Band-Pass Sigma-Delta Modulator for Mixed-Signal Systems-On-Chip

Thesis submitted as partial fulfillment of the requirements for obtaining the degree of "Doutor em Ciência da Computação"

Prof. Dr. Sergio Bampi
Advisor

Prof. Dr. Luigi Carro
Co-Advisor

Porto Alegre, julho de 2005.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Fabris, Eric Ericson

A modular and digitally programmable interface based on band-pass sigma-delta for mixed-signal systems on chip / Eric Ericson Fabris – Porto Alegre: Programa de Pós-Graduação em Computação, 2005.

116 f.:il.

Tese (doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2005. Orientador: Sergio Bampi; Co-orientador: Luigi Carro.

1.SOC. 2.Modulador sigma-delta 3.Interfaceamento reconfigurável. I. Bampi, Sérgio. II. Carro, Luigi. III. Título.

AGRADECIMENTOS

Ao longo dos últimos 5 anos, o presente trabalho de pesquisa foi construído com a participação de várias pessoas e instituições de forma direta ou indireta.

Eu gostaria inicialmente de agradecer a minha pessoa por ter tomado a decisão acertada pela qualificação acadêmica e à minha família por tê-la apoiado.

Aos meus orientadores e colegas o Prof. Dr. Sergio Bampi e o Prof. Dr. Luigi Carro pela competência técnica, determinação e paciência na orientação deste trabalho de pesquisa, a eles fica meu profundo agradecimento.

Um especial agradecimento às seguintes instituições:

- ao Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul por ter apoiado minha iniciativa através da concessão de afastamento das atividades de ensino e da infra-estrutura disponibilizada para o andamento deste projeto de pesquisa;
- ao Programa de Pós Graduação em Computação do Instituto de Informática da UFRGS pelo suporte dado através de seu corpo Docente, técnico e administrativo, bem como toda a infra-estrutura acadêmica e de pesquisa disponibilizada;
- aos órgãos do Governo Federal de fomento à pesquisa CNPq e CAPES, pelos financiamentos concedidos aos projetos de pesquisa que, de forma direta ou indireta, viabilizaram minha formação acadêmica e o desenvolvimento deste trabalho, bem como a publicação dos resultados dele decorrentes.

Aos meus pais um grande muito obrigado pelo incentivo, valorização e apoio dados antes e durante o desenvolvimento deste projeto pessoal de qualificação.

Finalmente, agradeço às pessoas que de alguma forma colaboram ao andamento na condução da atividade de pesquisa que resultou na conclusão desta tese de doutorado, mas que não foram citadas.

Dedico este trabalho à minha esposa Liliane pela sua compreensão e apoio, e ao meu filho Lucas pela sua alegria contagiante e seu jeito peculiar de pedido por atenção – ‘Papai posso desligar o seu computador?’.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

AD	Analog to Digital
ADC	Analog to Digital Converter
AEC	Analog Elementary Cell
AF	Audio Frequency
AGC	Automatic Gain Control
ASIC	Application Specific Integrated Circuit
BJT	Bipolar Junction Transistor
CAB	Configurable Analog Blocks
CMOS	Complementary Metal Oxide Semiconductor
CMRR	Common Mode Rejection Ratio
CT	Continuous Time
CTBPΔΣM	Continuous Time Delta-Sigma Modulator
DAC	Digital to Analog Converter
DR	Dynamic Range
DSP	Digital Signal Processing
DT	Discrete Time
FAC	Fixed Analog Cell
FDR	Frequency Dynamic Range
FPAA	Field Programmable Analog Arrays
FPGA	Field Programmable Gate Array
FPMA	Field Programmable Mixed Signal Array
IP	Intellectual Property
MF	Medium Frequency
MSO	Mixed-Signal Oscilloscope
OPAMP	Operational Amplifier
OSR	Over-Sampling Ratio
PF	Processing Frequency

PLL	Phase Locked Loop
PSD	Power Spectral Density
PSRR	Power Supply Rejection Ratio
RF	Radio Frequency
SC	Switched Capacitor
SFDR	Spurious Free Dynamic Range
SNR	Signal to Noise Ratio
SNDR	Signal to Noise and Distortion Ratio
SOC	System On a Chip
VCO	Voltage Controlled Oscillator
VGA	Variable Gain Amplifier

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The focus of this thesis is to discuss the development and modeling of an interface architecture to be employed for interfacing analog signals in mixed-signal SOC. We claim that the approach that is going to be presented is able to achieve wide frequency range, and covers a large range of applications with constant performance, allied to digital configuration compatibility. Our primary assumptions are to use a fixed analog block and to promote application configurability in the digital domain, which leads to a mixed-signal interface. The use of a fixed analog block avoids the performance loss common to configurable analog blocks. The usage of configurability on the digital domain makes possible the use of all existing tools for high level design, simulation and synthesis to implement the target application, with very good performance prediction. The proposed approach utilizes the concept of frequency translation (mixing) of the input signal followed by its conversion to the ΣΔ domain, which makes possible the use of a fairly constant analog block, and also, a uniform treatment of input signal from DC to high frequencies. The programmability is performed in the ΣΔ digital domain where performance can be closely achieved according to application specification. The interface performance theoretical and simulation model are developed for design space exploration and for physical design support. Two prototypes are built and characterized to validate the proposed model and to implement some application examples. The usage of this interface as a multi-band parametric ADC and as a two channels analog multiplier and adder are shown. The multi-channel analog interface architecture is also presented. The characterization measurements support the main advantages of the approach proposed.

Keywords: Sigma-Delta Modulators, Reconfigurable SOC Interface, System on Chip, FPAA, FPMA, high frequency analog operation.

Uma Interface Modular e Digitalmente Programável Baseada em Moduladores Sigma-Delta Passa-Banda para Sistemas em Chip de Sinais Mistos

RESUMO

O foco desta tese é a descrição e validação de uma arquitetura de interface para processamento de sinais analógicos para SOC de sinais mistos. A abordagem proposta apresenta a possibilidade de cobertura de uma larga faixa de freqüências com performance praticamente constante associada a uma estrutura digital de programação. A premissa é usar uma célula analógica fixa e promover a configuração da aplicação no domínio digital, levando a uma arquitetura de interface de sinais mistos. O emprego de um bloco analógico fixo busca eliminar a perda inerente de performance decorrente da própria estrutura de programação em circuitos reconfiguráveis analógicos. A emprego da programação no domínio digital abre espaço para usos da vasta gama de ferramentas disponíveis para o projeto em alto nível de abstração, simulação e síntese automática para implementar a aplicação alvo com excelente predição do desempenho final. A abordagem proposta baseia-se no conceito de translação em freqüência (mixagem) do sinal de entrada seguida pela sua conversão para o domínio ΣΔ. A estrutura de processamento possibilita o emprego de um bloco analógico constante, e também, um processamento uniforme de sinais de entrada indo de DC até altas freqüências. A aplicação é configurada no domínio ΣΔ onde a performance pode ser predita de acordo com as especificações alvo. Objetivando a exploração do espaço de projeto foi desenvolvido o modelo de performance teórico e de simulação. Os modelos desenvolvidos auxiliam no também no projeto físico da interface proposta. Objetivando, tanto a validação dos modelos propostos, bem como o desenvolvimento de aplicações, foram construídos dois protótipos. São apresentados os usos da interface como um ADC paramétrico multi-banda e como um multiplicador e um somador de sinais analógicos. É proposta também uma arquitetura para uma interface analógica multi-canal. Os resultados experimentais empregados para a caracterização da interface proposta suportam as vantagens da mesma.

Palavras-Chave: Reconfiguração analógica, modulador sigma-delta, mixer, FPGA, circuitos analógicos de alta freqüência.

1 INTRODUCTION

Silicon integrated devices have been experiencing continuous scaling down with CMOS technology advances what has enabled designers to include more functionality in the same integrated circuit. The demand for system-on-a-chip (SOC) applications is rapidly growing and creating the necessity of tools able to increase the degree of design automation and prototyping of such systems. It is common to find mixed signal SOC applications that deal with low and high frequency signals and signal processing functions (linear and non-linear).

For example, most of the engines in present days vehicles are digitally monitored and controlled. One set of sensors present in this system is a variety of temperature sensors. These signals must be conditioned and transmitted to a processor where the information is going to be used. Figure 1.1 sketches a block diagram of such application. For this case, some linear blocks such as amplifier, filters and an ADC are present. The whole system is controlled by a digital part comprising the SOC.

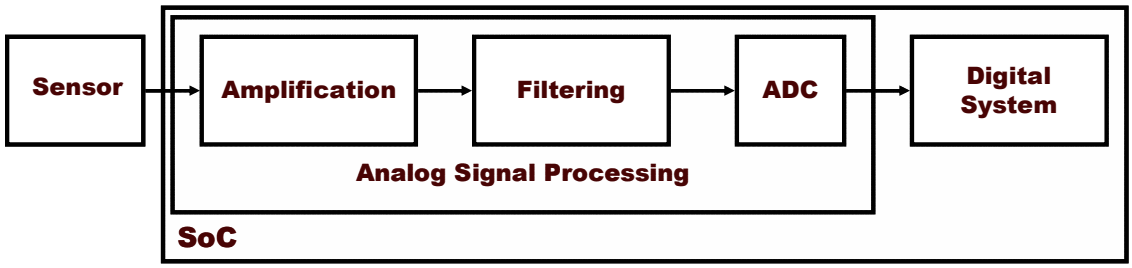


Figure 1.1: Example of a linear application of sensor conditioning within a SoC.

Another example where mixed signal SOC are present is the cellular phone. One can find some non-linear devices like a PLL that is responsible for frequency synthesis or signal recovery. Figure 1.2 shows a simplified block diagram of such non-linear application in nowadays mixed signal SOCs.

Many SOC applications require a mixed-signal (analog and digital blocks) design approach. Mapping these mixed-signal applications to a SOC demands a wide range of specific analog or mixed signal processing blocks, such as analog amplifiers, analog-to-digital converters (ADC), digital-to-analog converters (DAC), filters, mixers and other RF modules [LEV2002]. So, not only CAD tools have to support the mixed signal design environment, but the platforms used to carry these systems have to support mixed signal topologies. The incorporation of some degree of analog programmability is imperative in current digital reconfigurable devices or general purpose SOCs. Some initiatives like the **PSoC™** family from Cypress [CYP2003] and FIPSOC from Sidsa [SID2003] are emerging in the IC industry.

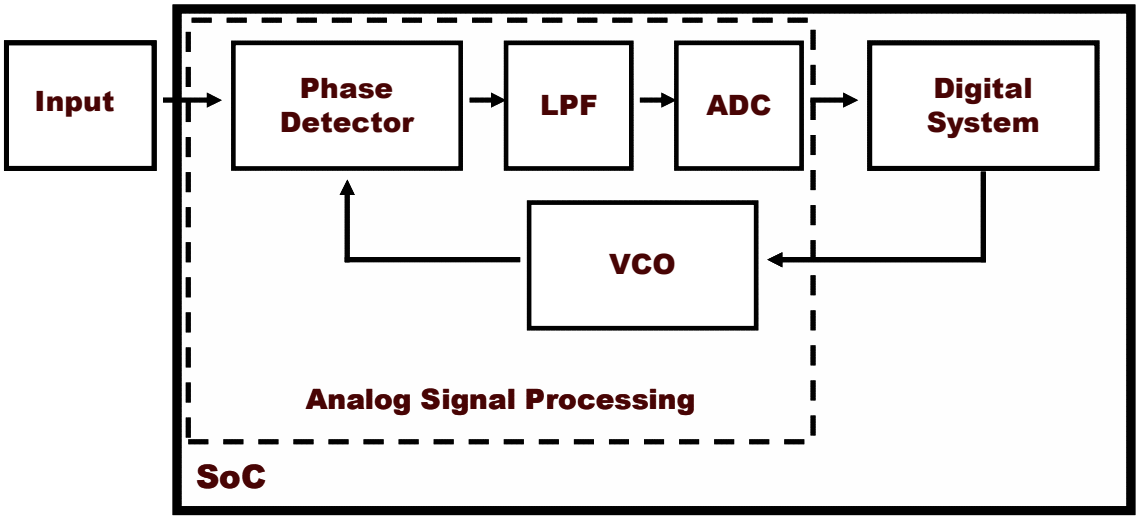


Figure 1.2: Example of a PLL nonlinear application within a SoC.

Nowadays, some multimedia ICs have to reach the market in less than 6 months [LEV2002]. Shorter development cycle and time to market with lower development costs make the reuse of parts already designed or the incorporation of third party intellectual property (IP) a mandatory solution. The reuse of digital blocks is definitely a much simpler task than reusing an analog IP. For the digital blocks, designers can rely on many EDA tools to rapidly plug the digital IP or a high level description to the target technology. However, the reuse of analog blocks has not yet reached this degree of automation or even a standard language description. Looking at the specification of analog blocks, one finds constraints as noise, thermal drift, stability, frequency response, linearity, power and others. This long list of parameters and their direct dependency on transistor sizing (technology dependency) and specific layout make the reuse task much more difficult. Often the reuse of analog subsystem needs some sort of tailoring to fit in the project constraints and technology specifications. This tailoring can go from transistor resizing and layout to some topological adjustments.

Researchers and manufacturers of EDA tools are still looking for a solution of analog and mixed-signal design bottleneck. One can classify already published approaches to analog design automation in two categories: design methods [GIN2002, SAH2002, SHI2002, SOM2002, PLA2001, WAN2002] and reconfigurable analog blocks [DME98, PAN2002].

Analog design automation is certainly lagging behind the digital counterpart. One of the reasons is that the design of an analog block is still largely based on transistor level modifications [PLA2001]. Some works on behavioral modeling (at different levels) are emerging to facilitate the top-down co-simulation of mixed digital and analog circuits [SOM2002]. The objective of these approaches is to obtain some independence from the target technology. The synthesis of analog blocks is mainly based on parameterized cells [SHI2002], where the analog function is mapped. The transistor size is determined with the use of some sort of optimizations [PLA2001]. SOC designers would greatly profit from ways to describe and to co-simulate digital and analog parts in a top-down fashion, to allow design space exploration, and also to automate the bottom-up verification process.

The other approach is based on reconfigurable analog devices that permit the fitting of analog functions with acceptable performance loss. In this category, the research that

has been done in industry and academia labs has proposed some architecture solutions for field programmable analog arrays (FPAA) and field programmable mixed signal array (FPMA) [DME98]. The search for small prototyping cost, short development time and reuse of analog blocks has guided many efforts towards providing accurate, low-cost and configurable circuits for analog and mixed signal prototyping [LEE98] [CHO95]. All proposed topologies rely on a set of configurable analog blocks (CABs) and some sort of interconnection network. One can find examples of continuous time and discrete time architectures, and programmability applied to linear or non-linear circuits, but usually not both in the same architecture.

Many times an FPGA platform is chosen not only as SOC prototyping environment, but as target final product carrier. Taking this scenario, a general analog interface able to manage the analog signal in nowadays SOC applications and FPGA platforms is highly desirable. This makes clear the necessity for an analog or mixed signal processing block architecture that could implement many applications, linear or nonlinear, with programmability and reuse targeted to current SOC designs.

The natural place to look for this interface solution should be among analog or mixed-signal programmable devices. The discrete-time FPAA and FPMA topologies are intrinsically limited to half the switching frequency, even though they are more suitable to digital control. A further limiting factor of discrete-time architectures is that they require an anti-aliasing filter. On the other hand, continuous-time architectures can achieve higher frequencies of operation, but a trade between flexibility and frequency response is necessary. As continuous-time designs are less suitable for digital control, they require a more complex configuration and interconnection structure. To improve the operating frequency, CT architectures demand a reduction in the interconnection matrix between configurable analog blocks, reducing application coverage.

All existing solutions for analog field programmable devices rely on the same configuration principle. The configuration of these devices is made by insertion of a configurable analog block in the analog signal path. This means that the most sensitive part of the system is changed to achieve the desire processing function. The configuration infrastructure always inserts some parasitic components (resistors, capacitors) and effects (crosstalk, noise, phase unbalance, charge injection). From this reality, there is an open space for research of an architecture that could deliver at the same time wide frequency range and cover a large application range with constant high performance.

The focus of this thesis is to propose and to model an architecture for interfacing analog signals in mixed-signal SOC. We claim that the approach that is going to be presented is able to provide wide frequency range and target application coverage with constant performance allied to digital configuration compatibility. Our primary assumptions are to use a fixed analog block and to promote application configurability in the digital domain, which leads to a mixed-signal interface. The employment of a fixed analog block avoids the performance loss common to configurable analog blocks [DME98]. The adoption of configurability in the digital domain makes possible the use of all existing tools for high level design, simulation and synthesis to implement the target application, with very good performance prediction. Our approach utilizes the concept of frequency translation (mixing) of the input signal followed by its conversion to the $\Sigma\Delta$ domain. This approach makes possible the use of a constant analog block, and also, a uniform treatment of input signal from **DC** to high frequencies. The application

is programmed in the ΣΔ digital domain where its performance can be closely predicted and achieved as initially specified.

This thesis is organized as follows. Section 2 discusses previous works towards analog programmability and reuse. Section 3 reviews some basic concepts and presents the proposed processing concept. The interface performance modeling is presented in section 4 and, in section 5, the prototype design steps are addressed with its characterization and application mapping examples. Sections 6 presents the conclusions and future works. There are also two appendixes one that explains the theoretical development shown in sections 4 and 5 and other that exposes the prototype debugging process.

2 ANALOG FIELD PROGRAMMABILITY: TECHNIQUES, ARCHITECTURES, AND DEVICES

D'Mello and Gulak in [DME98] provided a detailed survey of the field of programmable analog and mixed signal integrated circuits. Since the publication of the paper [DME98], few works have been published, but they are variants of some existing architecture [PAN2002]. It is possible to classify these ICs into two main purpose categories, one that deals with structural (functional level) programmability and the other one that implements parametric programmability. In the first group, the goal is to provide the device with different circuit topologies implementation possibilities like amplifiers, filters, adders, PLL and so on. The other group gathers the initiatives towards architectures for variable component values, variable gain amplifiers, AGC, configurable filters, etc.

An important characteristic that has direct impact on design and application instance is the operating time mode of the device. From this perspective, the device can operate in continuous or discrete time modes. The discrete time approach is more suitable for digital control, but it requires that the input signal must be band limited to half of the switching frequency. Hence, the use anti-aliasing (input) and reconstruction (output) filters is mandatory. On the other hand, the continuous time mode devices do not need a band limited input signal, but may require more complex configuration schemes to be digitally programmed [DME98].

Also it is important to state that there is a compromise between circuit programmability and frequency response. If a fully programmable topology is desired, a complex interconnection network is necessary. This complex interconnection imposes a restriction in the range of frequencies to be treated, because of the introduction of parasitic components that causes phase errors and cross talk that are critical, mainly in high frequency applications. This also makes the frequency coverage strongly dependent on the application. A solution for this problem is to limit the flexibility of the system, reducing the range of possible applications [PIE98a].

Regarding the configuration scheme, some approaches use metal-mask programmable arrays [GIR2003], standard analog cells or field programmable devices [DME98]. The first one is intended to be used in short fabrication cycle, where a pre-processed sea-of-transistors (**SOT**) or analog blocks are used to instantiate the desired analog function through the use of a custom set of metal masks. The analog standard cells are a challenging approach, since each target analog application has its own performance constraints (**DC**, **AC**, noise, thermal stability, etc.), which imposes a vast and parameterized cell library to guarantee functionality coverage. The field programmable devices approach has to provide not only the configurable analog blocks

itself, but also memory elements to store the configuration pattern and all interconnection infrastructure necessary to join the existing primitive components.

Figure 2.1 depicts the block diagram of a general FPAA architecture [DME98]. This general FPAA has a collection of configurable analog building blocks or CAB, a routing network and memory elements. The memory elements are responsible for retaining the information regarding interconnection elements and CAB. The input data port makes possible the configuration word to be transferred to each configurable node. According to the configuration pattern, each CAB will apply a processing function over its input. The tailoring of these processing functions is closely related to the configuration granularity of the CAB. For instance, the CAB can be composed of a set of elementary building block like differential pairs, current mirror, transconductors and so on, to macro blocks such as amplifiers, comparators, biquads, VCO, etc. Depending on the functionality delivered by the CAB and the signal routing, more complex functions can be applied over the input.

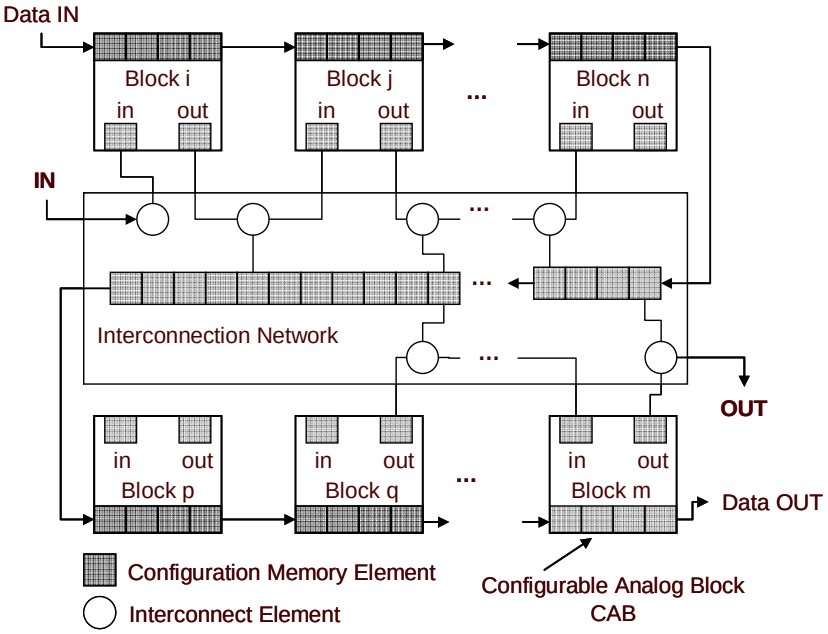


Figure 2.1: General architecture of an FPAA.

Depending on the approaches for analog programmability can be classified as figure 2.2 shows. According to the end user applications they are divided into two sets, one that permits only parametric programmability and the other one the whole functionality of the circuit can be reconfigured. The parametric programmable devices are those that some parameters are reconfigurable like gain in a **VGA**, the transfer function in configurable analog filter or resistance in digital potentiometers, for example. The other one group that allows structural programmability which the end user application can assume different functions as amplification, analog filtering, analog-to-digital or digital-to-analog conversion, and in some cases also map a non linear behavior like a **VCO**, a **PLL** or an oscillator, for example. The structural programmable devices group is also divided into two main sets, one that gathers the devices called *field programmable analog array* – **FPAA** and the other one composes the *field programmable mixed signal array* – **FPMA**. In the **FPAA** set, examples of discrete and continuous time topologies or both in the same device are found.

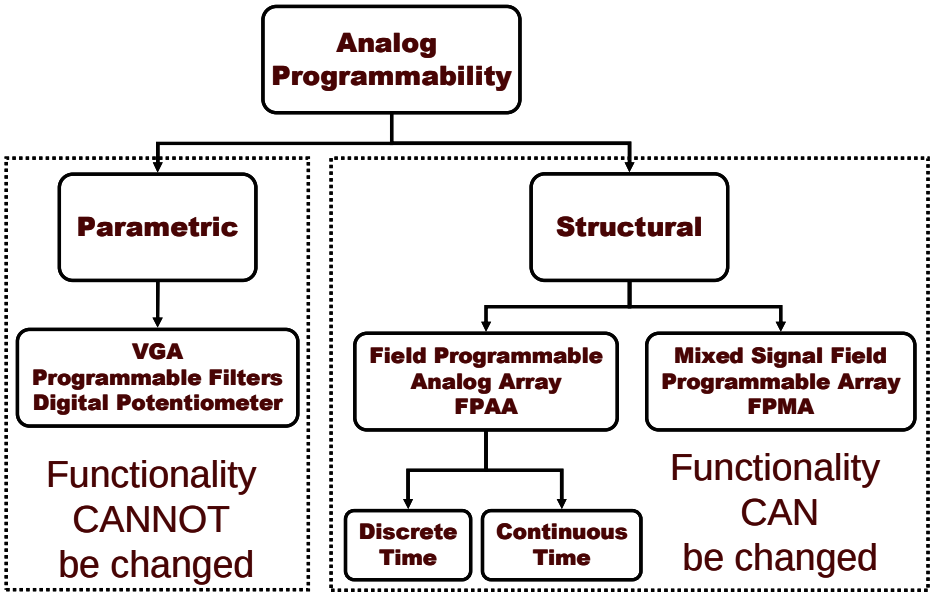


Figure 2.2: The analog programmability classification.

In the following, we analyze the existing solutions and architectures for analog programmable devices.

2.1 Analog structural programmability

In the literature, there are many approaches to provide topological reconfiguration. Analyzing these architectures it is possible to group then into two main sets the **FPAA** and **FPMA**, as described earlier. These two set are going to be reviewed in detail next.

2.1.1 Field programmable analog array

According to the time operating condition, the FPAA can be classified as continuous or discrete time architectures.

2.1.1.1 Continuous time architectures

The first use of FPAA came from the analog computers that were employed in the 1960s as hardware simulators in several fields of science and engineering [DME98].

It is intrinsic to continuous-time approaches not to restrict applications to band limited signal, but the circuit programmability may impose more complex topologies to achieve a large dynamic range. The insertion of programmable devices increases the overall noise decreasing the precision and frequency of operation of the input processing.

The area of test, synthesis and prototyping neural networks was also one of the driving forces to analog configurable structures, naturally targeted to implement programmable neural networks [CHA96]. Other approaches like programmable analog vector-matrix multipliers [KUB90] and a field reconfigurable IC called Sivilotti Proto Chip [SIV88] are reported with application also in neural networks and equation solvers. This class of application requires a variable gain amplifier (VGA) to adjust the neuron transfer function and interconnection weights.

Lee and Gulak proposed an FPAA where the interconnection switches were substituted by a variable resistor set implemented using a MOS-Transcondutor

composed by nine PMOS transistors. The building blocks chosen by the authors were OPAMP, MOS transconductor, capacitor and diodes. Target applications for this FPAA were general analog signal processing [LEE95, LEE92]. A fully functional prototype was built and some results were published reporting audio-frequency operation application.

In [PIE98a, PIE98b] a BJT FPAA approach is employed in high frequency applications. In this architecture, the authors proposed a cell array where the connection between cells was essentially locally made, with some limited global routing capabilities to reduce parasitic and reach high frequency performance. The interconnection and function programming used no switches in the signal path. The configuration task was performed changing the cells bias condition digitally through the use a collection of simple DACs. As the target technology was bipolar without MOS devices, the design of high quality analog memories was difficult. So, the analog memory cells were put in a ring structure and a refresh mechanism was used. Some simulation results of different classes of applications are shown including a PLL, a linear ladder filter with frequency of operation around 100 MHz. Low frequency of operation like in instrumentation is not reported.

Premont et al. [PRE98] proposed an FPAA architecture based on current conveyors. The main purpose of the paper is to introduce the current conveyors not only as basic building blocks to construct an analog elementary cell (AEC) to be used in a FPAA, but also as a switch element. Each AEC has two current conveyors and programmable capacitors and resistors. A transconductor implements the variable resistor. The variable capacitor is obtained using a fixed capacitor, two current conveyors and a variable resistor. Some examples of use in linear (full differential block, active filter, gyrator, etc.) and nonlinear (analog multiplier, frequency doubler, full wave rectifier) applications are addressed. The simulation results show that the proposed architecture achieve a frequency of operation around 1 MHz depending on application.

A current mode based FPAA is proposed by Embabi et al. in [EMB98]. In this paper, the authors remark that the first step into the design of a FPAA is to define the class of application it is intended for use. They focus the work to achieve a system that could be integrated with an FPGA to yield a field programmable mixed signal array (FPMA). A tradeoff is made between granularity and performance. The granularity of the configurable cell is adjusted to balance flexibility and degradation effects caused by the switching elements. Then, a restricted routing scheme is used. So, each cell can be connected to eight neighborhoods at most. To achieve high CMRR and PSRR, a fully differential structure is used in the configurable cell. The choice for the current mode approach comes from the fact that it is possible to achieve higher frequencies of operation and it is more suitable for low power supply voltage applications. Another aspect that the authors took in account was that the basic building for current mode architectures is the current mirror that can be easily implemented in standard CMOS technology. This, in turn, makes possible the integration of the proposed FPAA into a FPGA. The proposed architecture uses most switches in the biasing path of the configurable cell, this way reducing parasitic effects in the signal path. They show some simulation results and the target example is a band pass filter centered near 100 kHz.

The first attempt to release a commercial and monolithic IC with parametric and structural capabilities was made in the early 1980s by Precision Monolithics Inc. (PMI). The device called GAP-01 had two transconductance amplifier, one output buffer and an uncommitted comparator with the output voltage programmable through externals

resistors. The output of the transconductance amplifiers was selectable using a pair of digitally controlled current switches, with low glitch operation. Applications such as two channels sample and hold amplifier, an absolute value amplifier and a successive-approximation ADC were reported in its datasheet. [DME98].

Lattice™ Semiconductor [LAT2003] has an FPAA product line called ispPAC®. The ispPAC® product family has several members each one of them has some set of macro components intended for some specific application. For example, ispPAC20 has a collection of blocks like OPAMP and VGA, high speed comparators, an eight bit DAC, a VCO, etc. The main application of this device is for data acquisition front ends. Although the ispPAC81 has only a programmable active 5^{th} order low pass filter and a VGA, its main application is for precision reconstruction or anti-aliasing filter in the frequency range around tens of kHz.

2.1.1.2 Discrete time architectures

Discrete time architectures, with the switched-capacitor as a representative example, are more suitable for digital control. But, its sampling time nature intrinsically requires the signal to be band limited to at least a half of the switching frequency. Another factor for frequency response limitation is the signal routing, since the insertion of pass transistors and interconnection parasitic increase the time constant for charging and discharging the sampling capacitors, which contributes to decrease the maximum switching frequency. This technique also requires an anti-aliasing filter at the input and a reconstruction filter at the output, reinforcing the frequency limitation.

PMel-Motorola presented a FPAA based on switched capacitor technology [BRA96]. Figure 2.3 shows the basic architecture of the proposed CAB. The IC consists of 4x5 array of CABs each containing an amplifier, a variable capacitor array and a set of CMOS transmission gate switches. A RAM and a switch control manager inside each CAB are responsible for holding the configuration pattern and for controlling the connectivity and switching phase to implement the desired function. A CAB can be used to implement an amplifier, a comparator, first order filter sections, integrators and differentiators. Applications reported are full wave rectifier, a PCM CODEC and a VGA. A commercial product was released by Motorola in 1997. In January 2000, as venture-backed technology spin-off from Motorola, Anadigm® [ANA2003] was founded. The basis of their FPAA product line is this architecture. They incorporate some extra functionality like dynamic reconfiguration. Maximum operating signal bandwidth is 2 MHz and it is highly application dependent.

An Electrically Programmable Analog Circuit (EPAC™) is introduced in [KLE98]. This design is intended for commercial use. The approach was to give a restricted access to low level components, this way making visible to the user only some programmable macro blocks, having almost the same concept as used in the FPGA parameterized macro functions. Some applications are shown with signal maximum frequency around few hundreds kHz.

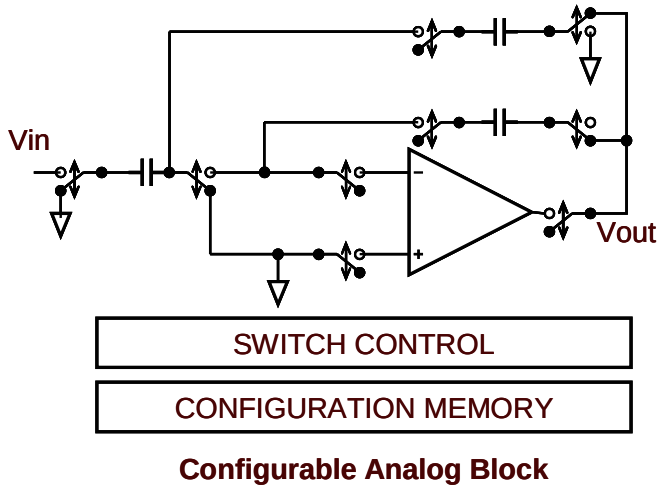


Figure 2.3: PMeL-Motorola FPAA analog cell block diagram.

A switched capacitor FPAA using lossless integrators and stray insensitive architecture was proposed by Kutuk and Kang [KUT98], as shown in figure 2.4. The maximum signal frequency is limited to 125 kHz for this implementation. Controlling the switching scheme inside the CAB they achieved an inverting and non inverting integrator. Since the CAB is a second order topology, it is possible to implement a large number of biquad filter functions controlling the switching mechanism. Applications such as filters, amplifiers, modulators and signal generator have been reported.

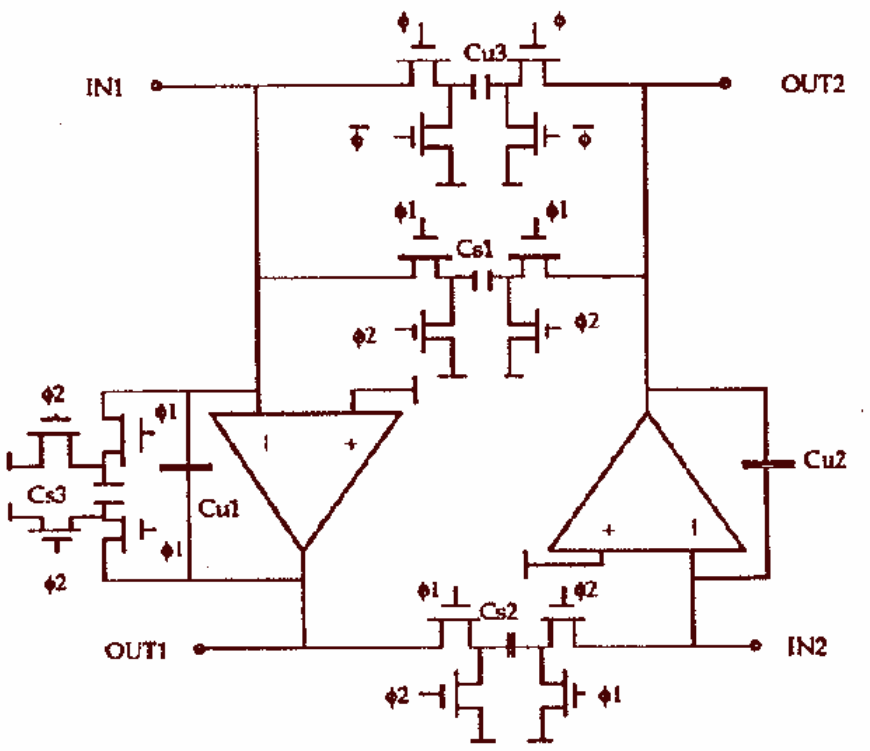


Figure 2.4: Switch capacitor CAB [from KUT96].

Other architectures variants of switch capacitor CAB and interconnection are reported like in the work of Lee and Hui in [LEE98] and Bratt and Macbeth in

[BRA98]. These works focus in new arrangements for the CAB and the interconnection infrastructure looking for improving application coverage and frequency range.

2.1.2 Mixed signal field programmable analog array

The basic topology of a FPMA is the union of the digital and analog infrastructure through the use of a set of ADC and DAC. Figure 2.5 illustrates this concept. The whole idea is to conjugate the resource of digital and analog array to build an infrastructure able to process signals (digital, analog or both) according to a prescribed function. The work of Chow et al. [CHO95] introduced this concept of bringing to work together the FPGA and the FPAA devices. Their work analyzes a collection of target applications to balance the size of analog and digital array. The results showed that the number of connections into digital or analog array was bigger than the interconnection between the digital and analog parts. From this preliminary study, a prototype called MADAR was fabricated. The converters were implemented by using a custom configurable converter structure, although it was shown that was possible to build the converter blocks using existing resources in the digital and analog arrays. This decision for a customizable converter was taken based on area overhead and low speed of operation when they were implemented by existing resources in the FPGA and FPAA. The maximum clock frequency reported is 1 MHz for the dual slope converter example giving a sampling frequency of 62.5 kSamples/s and resolution of four bits.

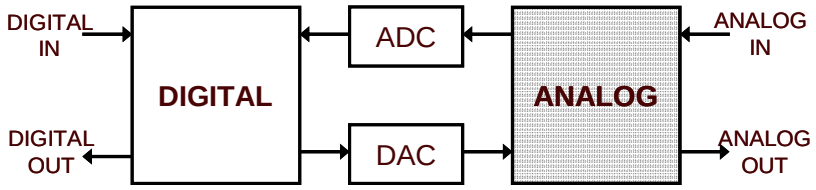


Figure 2.5: Conceptual view of a FPMA [from CHO95].

Faura et al. designed a Field Programmable System On Chip (FIPSOC) that is composed by a coarse-grained lookup-table based FPGA, a reconfigurable analog block and an 8051 microprocessor core, as shown in figure 2.6. This device was aimed to reduce the development time of systems with digital, analog and software components [FAU97]. The CAB in the FIPSOC chip has four channels with features like variable gain, filtering features and comparison capability, with parametric only programmability. That is, the CAB cells have fixed functionality. The interface between digital and analog parts is made through reconfigurable resolution ADC (one 10 bit or two 9 bits or four 8 bits) and DAC, each data conversion block can be independently configured as a DAC or ADC block with maximum conversion rate of 250 kHz (in pipelined mode could achieve 1 MHz). An internal routing matrix makes possible for the processor the acquisition of almost all internal analog nodes. Another feature of this design is a double buffered configuration memory set that makes possible on the fly reconfiguration of the device by the microprocessor.

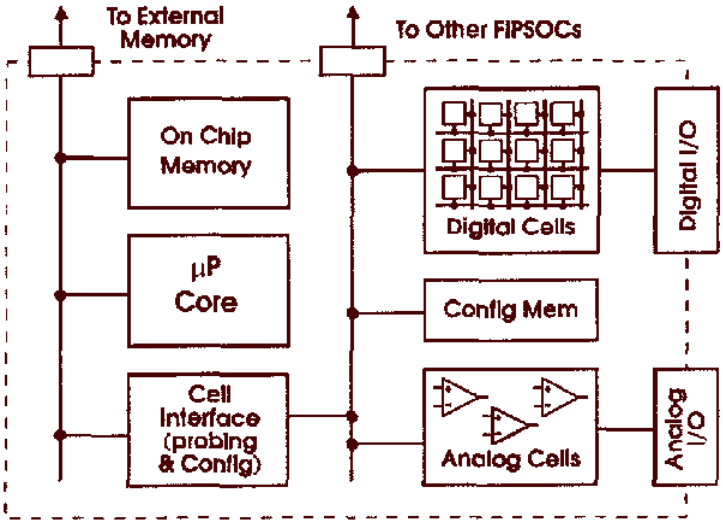


Figure 2.6: Building blocks of FIPSOC from Faura et al. [FAU97].

In 2000 Cypress® Semiconductor Inc [CYP2003] introduced the PSOC® family of programmable system on chip microcontrollers. The PSOC® product line concept follows the same basic principles proposed by Faura et al. [FAU97]. The architecture of these devices is composed by a microcontroller core of 8 bits, program and data memory blocks, a set of configurable digital (eight) and analog blocks (twelve). The difference between FIPSOC and PSOC concerning the analog cells is that PSOC® CAB cells permit structural configuration, this way making possible the implementation of a larger set of functions. The analog part, for instance, is a collection of continuous and discrete time (SC) blocks. The analog blocks arrangement of PSOC® CY8C2XXX family devices has twelve analog configurable basic blocks, all continuous time (CT) are identical, but the discrete time (DT) ones are of two types. A complex routing mechanism is available for signal routing and block configuration. The blocks are organized in matrix format with the CT blocks on the top of each column. Looking at the datasheet it is possible to see two classes of applications, one for the CT blocks and other one for the SC blocks. The user can map a variety of signal conditioning and detection (VGA and comparator) in the CT blocks and data conversion and filtering in the SC blocks. It is reported data conversion resolution up to 12 bits for the ADC and 9 bits for the DAC, depending on the architecture (incremental, ΣΔ or successive approximation) of the implemented converter. The digital configurable blocks can be used to implement the converter controller part. From the technical data, the analog front end can properly work with signals up to few hundreds kHz, depending on the application. The gain (attenuation) achievable in each CT block is 16 (0.0625). A precision voltage reference is also available.

2.2 Analog parametric programmability

The devices in this group permit the configuration of some parameters of their functionality but do not permit change in their functionality itself. An amplifier with variable gain (VGA), with which the user can only change gain or attenuation but cannot use it as comparator is an example of this class of IC. It is not the scope of this review to go deep inside this class of devices, so only a general overview is going to be made to show the difference with previous approaches.

Xicor® [XIC2003, DEM98] introduced in the early 1990s an IC that implemented a digital potentiometer. The wiper position was digitally controlled using an UP/DOWN counter, so the user could control it using push buttons. These devices can have linear or log scale with different resistance values, but not in the same device. This component class made possible the development of commercial products with board level analog digital programmability. Now, Xicor also offers digitally controlled capacitors.

Another member of this class is the *variable gain amplifier* – **VGA**. The first commercial digitally programmable VGA devices were introduced by Burr Brown Inc (now part of Texas Instruments Inc). Those IC had their use into data acquisition systems for instrumentation. Analog Devices [ADI2003] has also VGA for the field of RF/IF, video automatic gain control and measurements applications. Mostafa et al. published in 2003 [MOH2003] a CMOS VGA with 60 dB of gain selectable in 2 dB step with operating frequency of 246 MHz for GSM application having an operating bandwidth of 5 MHz.

There are several commercial programmable switched capacitors (discrete-time) or continuous time filters from many manufactures [ADI2003, ANA2003, ONS2003, DAL2003, NAT2003, DME98] available. In some of them the configuration is digital compatible, but others require external passive components are involved to achieve programmability.

2.3 Discussion

Independent of the circuit granularity, the programmability of all FPAA and FPMA reported in the literature and industry is achieved by changing the topology of a CAB, through the use of a complex network of pass-transistors for routing or switching signals (currents or voltages), capacitors and resistors or transconductors. In order to reach the required programmability, all analyzed implementations take the same starting point, that is, one must change the behavior of the analog circuit in order to match the desired processing function. This way, one must always change the most sensitive part of the design, the analog part itself. In contrast with their digital counterparts, reconfiguration does not mean only that the circuit may become slower, but several additional parameters could be affected off-set voltages, noise figure, cross-talk and so on. In the analog domain, this means the circuit might not work at all. High frequency filters and mixers are a clear example of this problem.

The linear applications are mainly filters and amplifiers. Non-linear applications are restricted to a set of few circuits, such as comparators, multipliers, rectifiers and, in some cases, voltage controlled oscillators. It is also reported the implementation of low frequency analog-to-digital and digital-to-analog converters. Clearly, the reported FPAAs are limited in the frequency coverage, as well as in the application range. This complex interconnection imposes a restriction in the range of frequencies to be treated, because of the introduction of phase errors and cross talk that are critical in high frequency applications. A solution for this problem is to limit the flexibility of the system, reducing the range of possible connections, and hence the range of applications.

All presented architectures have a common bottleneck, that is, the performance penalties caused by the same devices or structure that make the reconfiguration possible. Whenever extra hardware must be inserted, extra parasitic capacitances and extra switches are involved in the signal path. While in the digital domain the extra switches and paths mean a slower circuit, in the analog domain some circuits will

simply not work at all, like a high frequency filter or a mixer, for example. This way, there is not much hope of extending the performance capabilities of FPAA to the same level, or even to a close range of their dedicated circuit counterparts. Table 2.1 summarizes the research at academia and industry towards analog field structural programmability, where **AF** – Audio Frequency range (DC - ±100 kHz), **MF** – Medium Frequency Range (above 100 kHz up to a few MHz) and **HF** – High Frequency range (above 10MHz).

Table 2.1: List of the main published approaches for structural analog programmability and SOC interface solutions.

Author	CAB Technology	Freq. Range			Application Mapping Reported	Results
		AF	MF	HF		
Lee and Gulak [LEE98]	CT CMOS	■			Filter	Full Implementation
Pierzchala and Perkowski [PIE98]	CT BJT			■	Filter, amplifier, PLL, VCO	Simulation / Measurement
Premont, et al. [PRE98]	CT CMOS	■	■		Diff. amp., filter, analog multiplier, rectifier	Simulation
Embabi, et al [EMB98]	CT – MS CMOS	■			Filter, comparator, pipelined ADC	Simulation
ispPAC® [LAT2003]	CT CMOS	■	■		Analog front-end, filter, VGA	Commercial
Pankiewicz [PAN2002]	CT	■	■		10 MHz basic block, filters (from few kHz to few MHz)	Prototype fabricated.
PMel-Motorola [BRA96]	DT – SC CMOS	■	■		Filter, PCM codec, rectifier.	Full Implementation
EPAC [KLE98]	DT – SC CMOS	■			Filter, sig. cond., comparator	Full Implementation
Kutuk and Kang [KUT98]	DT – SC CMOS	■			Amplifier, filter, signal generator, modulator	Simulation
Lee and Hui [LEE98]	DT – SC CMOS				ΣΔ modulator, AM modulator, filter, ADC.	Simulation at topological level
DPAD2 [BRA98]	DT – SC CMOS	■	■		BW 500 kHz, SNR 60dB, Filter, rectifier, amplifier.	Full implementation
MADAR [CHO95]	MS CMOS	■			Configurable ADC, comparator, analog front-end	Full implementation
FIPSOC [FAU97]	MS CMOS	■	■		Analog data acquisition front-end, conf. ADC.	IP from SIDSA [SID03]
PSOC® [CYP2003]	CT – SC – MS CMOS	■			Analog data acquisition front-end, filters, conf. ADC, signal generation, VGA	Commercial product

From this review, we conclude that there is an open space for research in the direction of one analog interface architecture able to deal with linear and non-linear applications with large range frequency coverage allied to digital **CMOS** compatibility for taking full benefit of Moore´s law as shown by figure 2.7. This interface should be able to deliver the following attributes:

- performance almost constant from DC to high-frequencies for band limited signals, this way covering many applications;
- capacity of realization of a wide variety of linear and non-linear applications;
- ability to balance between SNR and bandwidth according to the applications constraints;
- compatibility with digital control and programmability;

- CMOS technology compatible, to be in the industry main stream.

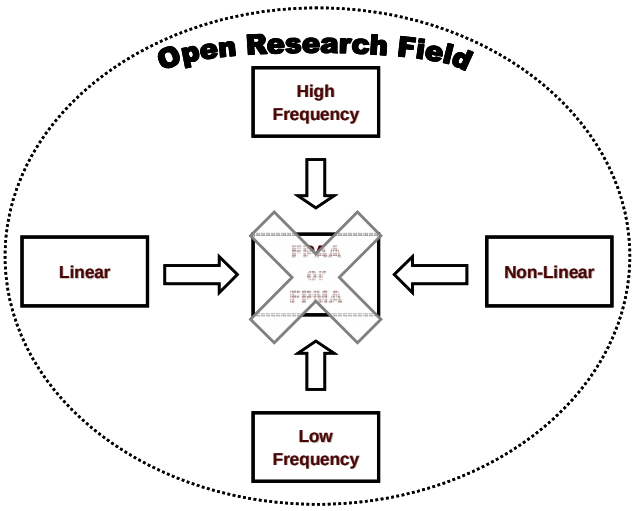


Figure 2.7: The heterogeneous application mapping over an **FPAA** or **FPMA**.

Indeed, a new design paradigm is needed to fulfill the above requirements. Hence, we propose a new look at the analog field programmability. The goal of this thesis is to present a new architecture that is able to overcome the performance limitations of present days FPAAs and FPMAs. The objective is to provide a programmable circuit that can maintain its performance over a large band of frequencies, and can be tuned to allow linear or non-linear processing functions reconfiguration. In order to achieve this goal, we propose the use of a fixed analog part that can be built with the best techniques to achieve high performance. Analog signal processing programmability is done by the change of digital parameters, with minimal performance degradation.

3 A SOC GENERAL ANALOG INTERFACE

All reviewed techniques used for structural or parametric analog configuration propose some circuit modifications to achieve a certain analog function. Taking into account that for most applications the input signal is band limited, one could think of reallocating this input signal to a fixed frequency band in the spectrum, so that a fixed and high performance mixed-signal section could process this translated version of the input signal. The concept of frequency translation is not new, and the use of intermediate processing frequency is a well known mechanism in telecommunication [LAT98] and in some instrumentation applications like chopper and lock-in phase amplifiers. However, the use of this theory to general signal processing is really new. This mechanism makes possible to process signals from DC to high frequency using the same infrastructure.

3.1 Frequency translation

The frequency translation mechanism can be derived from Fourier theory. If one multiplies two signals in the time domain, the resultant spectrum is equal to the convolution of the two spectra, one from each signal, like proposed in equation (3.1) [LAT98].

$$v_o(t) = v_i(t) \times v_{LO}(t) \Rightarrow V_o(jw) = V_i(jw) * V_{LO}(jw) \tag{3.1}$$

To illustrate the frequency reallocation, let us take the signal $v_i(t)$ and $v_{LO}(t)$ as given in (3.2) and apply in (3.1).

$$\begin{aligned} v_i(t) &= A_i \cos(2\pi \cdot f_i t + \Phi_i) \\ v_{LO}(t) &= A_{LO} \cos(2\pi \cdot f_{LO} t) \\ v_i(t) \cdot v_{LO}(t) &= \frac{A_i \cdot A_{LO}}{2} \cos(2\pi \cdot (f_i - f_{LO})t + \Phi_i) + \frac{A_i \cdot A_{LO}}{2} \cos(2\pi \cdot (f_i + f_{LO})t + \Phi_i) \end{aligned} \tag{3.2}$$

Figure 3.1 shows the modulation process in the time and frequency domains. The result is the input signal $\mathbf{v_i(t)}$ translated to two different positions in the resultant spectrum. These two frequencies are equal to $\mathbf{f_i+f_{LO}}$ and $\mathbf{f_{LO}-f_i}$. Setting the low frequency image $\mathbf{f_{LO}-f_i}$ as the constant frequency $\mathbf{f_p}$ where the signal should be acquired and processed, equation (3.3) states how the local oscillator frequency is determined to bring the input signal to the fixed processing band.

$$f_{LO} = f_i + f_p \tag{3.3}$$

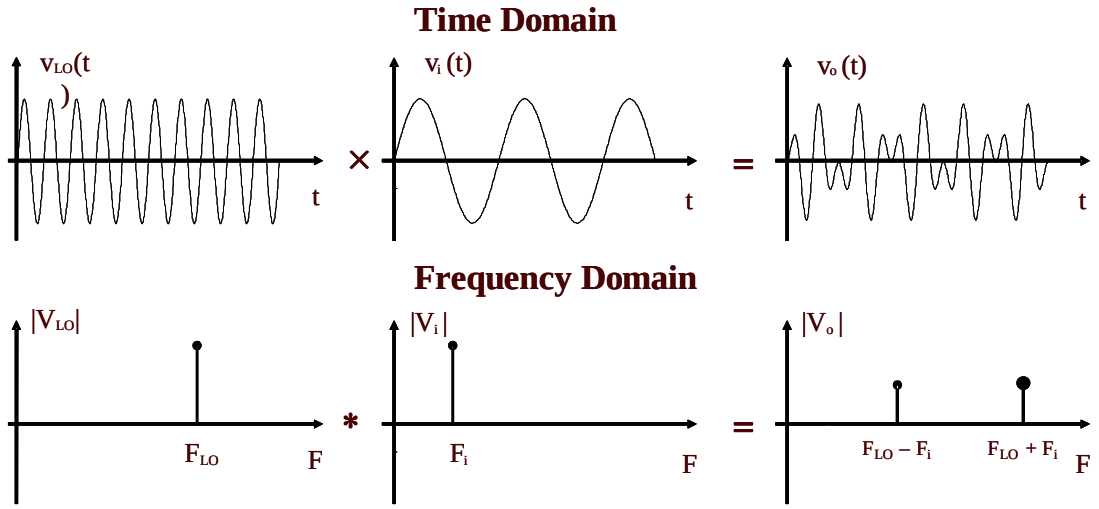


Figure 3.1: Frequency translation process.

The analysis developed above assumed the input signal as a single frequency. The results can be extended to a band limited signal with bandwidth **BW**, with central frequency equal to f_i. Now, each one of the images is centered at the frequencies f_i+f_{LO} and $f_{LO}-f_i$ with the same original bandwidth, exactly the same way as in the amplitude modulation technique [LAT98].

However, in order to allow easy generation of the local oscillator signal, one must wonder what happens if the waveform of the local oscillator is a square wave instead of a sinewave. Since the square wave is a periodic signal with frequency equals to $\mathbf{f_{sw}}$, a Fourier series can be used to express it as shown in expression (3.4) [LAT98].

$$v_{sw}(t) = \sum_{n=1}^{\infty} a_n \sin\left[2\pi(2n-1) f_{sw} \cdot t\right] \qquad (3.4)$$

$$a_n = \frac{2}{\pi(2n-1)}$$

Expression (3.4) shows that the square wave is composed by an infinite set of cosine functions. These cosine functions have frequencies that are equal to the fundamental frequency of the square wave and its odd multiples. Equations (3.4) also states that the amplitude of each harmonic decreases as the frequency increases. Mixing the desired input signal with a square wave gives as resultant spectrum the convolution of the two spectra, as shown by expression (3.1). The convolution in frequency creates an infinite number of images of the input signal and the position of each image is given by expression (3.5).

$$f_{img} = (2n-1) f_{sw} \pm f_i \quad \text{with } n = 1,2,3... \qquad (3.5)$$

In the case of using a square wave, the lowest image that corresponds to **n** equal to 1 in equations (3.4) and (3.5) is selected as the center frequency $\mathbf{f_p}$ of the processing band. This image is selected because it has the highest translation frequency gain, as shown in expression (3.4).

To separate the correct image (to be processed) from the others, a band pass filter with its central frequency set to the desired frequency image has to be used. The choice of the central frequency has to take into account several aspects, as:

- frequency coverage – for a given frequency span of the input signal, the lower is the selected frequency position, the higher is the requested frequency dynamic range of the local oscillator, as stated by expression (3.3);
- processing bandwidth (**PBW**) – for a given **PBW,** moving the selected center frequency to higher frequency requires a filter with higher equivalent Q; on the other hand, lowering the center frequency requires a higher order filter, so there is a tradeoff that might be taken into account;
- ADC sampling rate – the higher is the processing frequency $\mathbf{f_p}$, the highest is the required sampling rate.

3.1.1 Concept of mixed signal processing using frequency translation

The starting point of this development was to look for a way to overcome the penalties that arise from reconfiguring analog parts in a FPAA and FPMA. So, we started looking for an architecture concept that could make possible configurability with large frequency coverage, controllable losses, and capability to implement a wide spectrum of applications. A secondary aspect taken into account was related to available tools for design automation. This way, a natural domain to carry out the configuration is the digital domain. The digital domain not only has the desired support by the design automation tools, but also the technology for configurable devices is well established. Moreover, the digital domain could be valuable to process the input signal because all tools developed for digital signal processing could be used for that purpose. In addition, the programmability could be performed by a FPGA section, what could overcome the problem of reconfiguring the analog circuit itself, which is the most sensitive to performance loss. With these constraints, we concluded that a paradigm change was necessary.

It is important to observe that in almost all the practical HF cases, the HF signal occupies a small fraction of the band when compared to its central frequency, for example in AM and FM broadcasting telecommunication signals. So, the input signal HF center frequency is the hardest limitation and not the signal bandwidth, which is a fraction of percent of the local oscillator, when considering a digitizing infrastructure. This way, the approach based on frequency translation permits the use of an ADC with lower sampling rate than would be required to directly digitize the original HF signal. The use of a mixer in front of the data acquisition subsystem enables the manipulation of band limited signals located in a wide range of the spectrum, covering from DC to high frequency.

The conjugate use of a digitizing infrastructure, a mixer and a configurable digital block could be used as a framework for general analog signal processing. The input and output front-end could be fixed. The desired processing function programmability is achieved by the conversion to digital domain of the mixed input signal, and processing it with a configurable digital block as an FPGA. This approach allows the processing of the input band limited signal with a fixed and homogenous analog interface. Hence there is no performance penalty caused by the change in the analog subcircuits. The configuration of the digital part has a predictable behavior and all available tools for digital description, simulation, synthesis and hardware mapping can be directly used.

By using an intermediate frequency $\mathbf{f_p}$ (far away from DC) to acquire the signal, instead of working at its baseband, one can attenuate such problems as the 1/f or flicker noise, DC offset and DC offset drifts that are troublesome in the baseband. Moreover,

the same processing mechanism allows us to process signals from DC up to HF, which would not be possible with any of the known FPAA or FPMA topologies.

Figure 3.2 depicts the block diagram of this signal processing framework for comprehension purposes. The frequency of the local oscillator is determined according to (3.3), and from the spectrum location of the desired input signal. The input signal is mixed with the local oscillator to reallocate its spectrum. The mixed signal passes through a band pass filter to select the appropriate image. The selected image is sampled and digitized by an analog-to-digital converter (ADC). The sampled data is processed by a configurable digital system and converted back to analog domain using a digital-to-analog converter. The digital block demodulates the signal and processes it in its base band. The desired output signal is sent out by the output block composed by a DAC and a reconstruction filter.

To illustrate how to process an input signal using the proposed approach, the application of a filter over an input signal was simulated using Matlab®. The simulation results show the use of this technique to implement a Butterworth low pass filter over an input signal near DC (center frequency $\mathbf{f_i}$ = 0 Hz). The center frequency $\mathbf{f_p}$ of the processing band was set to 10 MHz. The image select band pass filter was a second order filter with Q of 40 corresponding to a bandwidth of 250 kHz. The sampling rate was set to 40 MS/s. The input signal was composed by three cosines with amplitude of 1V and frequencies equal to 1 kHz, 4 kHz and 10 kHz. The filter to be applied over the input was designed to have a cutoff and a stop frequency at 1.5 kHz and 4 kHz, with attenuation equal to 1dB and 60dB, respectively, figure 3.3 sketches this specification. These requirements lead to a Butterworth filter of 8th order. Figure 3.4 shows the simulation results.

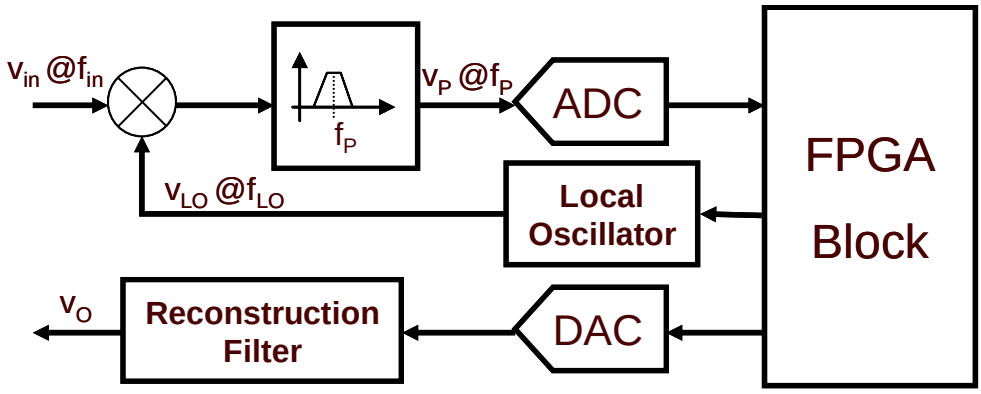


Figure 3.2: A mixed signal interface structure for SOC based on frequency translation.

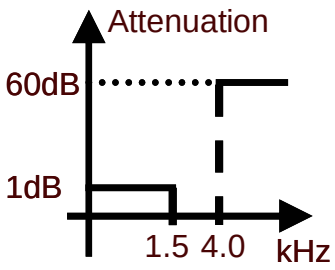


Figure 3.3: Filter frequency specification.

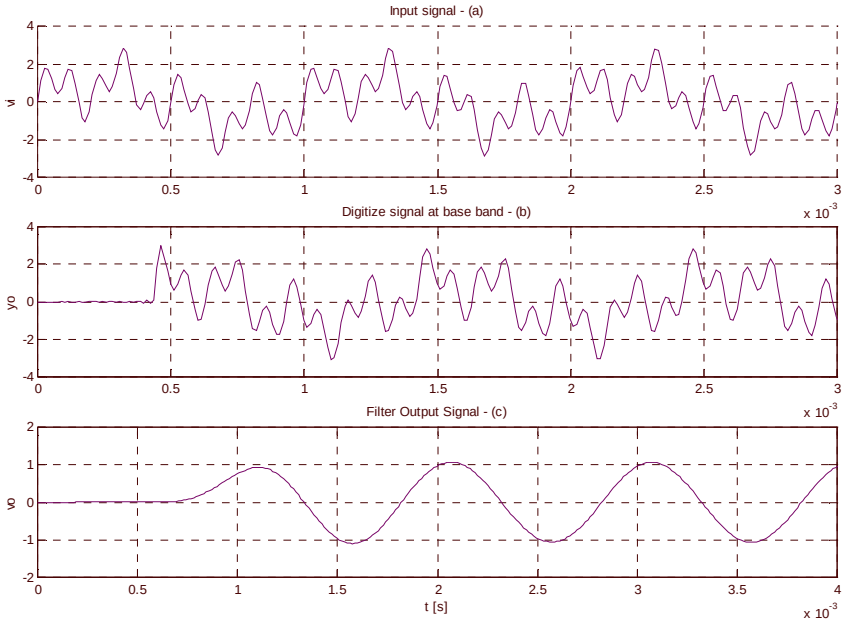


Figure 3.4: Simulation result of applying a Butterworth of order 8 over an (a) input signal, (b) the digitize version of input signal at base band and (c) output of processed signal.

Although high performance mixed signal processing can be achieved by the use of our reconfigurable structure, this certainly has drawbacks. Besides the area penalty, the power dissipation is much higher than a dedicated analog circuit. Low frequency CMOS analog processing can be performed at very low currents, even in the moderate-to-low inversion regime. The proposed programmable architecture uses intensive signal processing developed at a high sampling rate. On the other hand, digital FPGA are also power-hungry when compared to an **ASIC** doing the same function.

The use of conventional Nyquist rate analog-to-digital and digital-to-analog converters brings some additional disadvantages. The weakness of the first one regards the realization techniques and the target process technology. The performance of this kind of converter relies on the accuracy of its implementation, sometimes not compatible with the quality of required components available in conventional digital CMOS technologies. A further limitation comes from their working as sampled-data system. From this nature two requirements come: a sample-and-hold amplifier and an anti-alias filter [NOR97]. The Nyquist rate converters need a band limited input signal to avoid alias. In our approach, the band-pass filter in figure 3.2 must provide the channel selection and anti-alias functionality at same time. The converter bandwidth must be compatible with working center frequency $\mathbf{f_p}$, so the converter bandwidth naturally must be larger than signal bandwidth.

To avoid the above mentioned drawbacks, we proposed to substitute the conventional Nyquist rate converters by over-sampled sigma-delta modulators. The proposed replacement attenuates the requirements of the sample-and-hold amplifier and anti-alias filter [NOR97], and provides the ΣΔ domain as complimentary domain for mixed-signal processing with more efficiency [DIA95]. Moreover, the employment of over-sampled ΣΔ modulator as a conversion framework creates a homogeneous

environment for signal treatment with conventional CMOS technology compatibility. This approach will be addressed in the following subsection.

3.2 The ΣΔ domain as a signal processing framework

The ΣΔ modulator trades resolution in time for resolution in amplitude, such that the use of imprecise analog circuits can be tolerated. This relaxed constraints in analog circuit blocks performance makes this technique suitable for implementation in conventional digital CMOS technologies. Although commercial sigma-delta A/D and D/A converters have been in existence for more than two decades now, the primary application of such converters has been in digital audio. The narrow bandwidths in digital audio applications have made over-sampled converters particularly appealing. It is only recently, as we benefit from the increased speed of submicron devices, that sigma-delta modulators are exploited for wider band systems such as wireless RF communications [TAO99b, NOR97].

The ΣΔ data converters have proven their enormous flexibility for many applications requiring data acquisition ranging from sensor monitoring to high frequency telecommunication systems [VAZ2003]. Many researchers have shown additional advantages in using the ΣΔ modulated bit-stream as a processing signal domain [DIA95]. Also, the ΣΔ domain is an interesting compact signal representation, making it possible the design of many applications in more efficient ways than in conventional word-wide implementations. This efficiency comes from the hybrid nature of the ΣΔ data bit-stream. The result of passing an input signal through a ΣΔ modulator using a single bit quantizer (comparator) is a digital bit-stream. This bit-stream carries the input signal information buried in some quantization noise. Therefore, the ΣΔ bit-stream can represent, in a single bit compact form, an analog (or multilevel) signal. Moreover, Dias in [DIA95] proposes to consider this compact form of representation as an additional domain for signal processing. The use of ΣΔ domain as signal processing domain comes from its resemblance to continuous-time, sampled-data and digital domains at same time.

In this section, we are going to briefly review the basic signal processing in the ΣΔ domain, the metrics used for ΣΔM and the architecture of a framework for configurable signal processing using ΣΔ modulators and ΣΔ domain.

3.2.1 The sigma-delta modulators

Quantization of amplitude refers to the mapping of a continuous amplitude signal to a finite number of discrete levels and is at the heart of all digital converters and modulators. The difference between the original continuous amplitude and the new mapped value represents the quantization error. Figure 3.5 illustrates the quantization process. Qualitatively, it can be observed that the quantization error gets smaller as the number of discrete levels increases. The number of levels is in turn proportional to the resolution of the quantizer used in the ADC. Increasing the quantizer resolution will decrease the quantization error. The error is a strong function of the input; however, if the input changes randomly between samples by amounts comparable to or greater than the spacing of the levels, then the error is largely uncorrelated from sample to sample and has equal probability of lying anywhere in the range of $-\Delta/2$ to $+\Delta/2$ [NOR97]. The quantization step **Δ** is defined by equation (3.6), where **FS** represents the input full scale range and **B** the number of bits of the converter. Further, if it is assumed that the error

has statistical properties which are independent of the signal, the error can then be represented by a white noise.

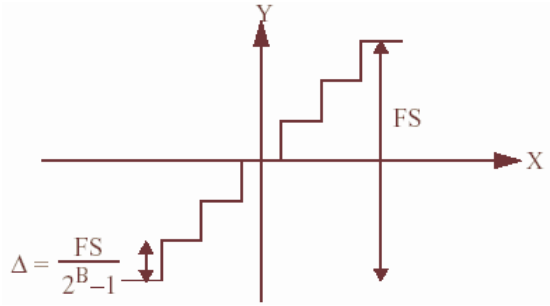


Figure 3.5: Quantization error in an analog to digital converter.

$$\Delta = \frac{FS}{2^B - 1} \tag{3.6}$$

The quantization noise is given by the mean-square value of the quantization error described above. Using a double-sided spectrum, the quantization noise is given by expression (3.7) and it is assumed to fall between $-f_s/2$ and $f_s/2$ where $\mathbf{f_s}$ is the sampling frequency. In Nyquist rate ADC this frequency is normally slightly greater than twice the signal bandwidth.

$$P_{Q,Nyquist} = \frac{1}{\Delta}\int_{-\Delta/2}^{\Delta/2} q^2 dq = \frac{\Delta^2}{12} \tag{3.7}$$

In over-sampled data converters, the sampling frequency is much higher than the signal bandwidth ($\mathbf{f_{BW}}$). From equation (3.7), the quantization noise power is independent of the sampling frequency. As the quantization noise is the same for the Nyquist rate and over-sampled ADC, the noise power density for the second is lower than first one as shown in figure 3.6.

The in-band quantization noise power, the shaded region in figure 3.6, is expressed by equation (3.8). Therefore, if the over sampling ratio (OSR) is incremented by a factor the in-band noise power decrease by the same factor.

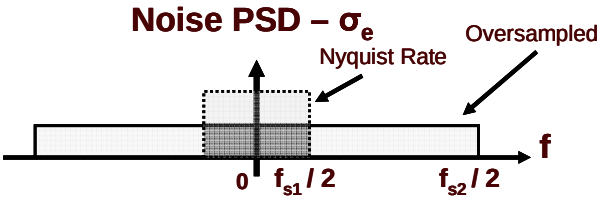


Figure 3.6: Quantization noise in Nyquist-rate and oversampled ADC.

$$P_{Q,Oversampled} = \frac{P_{Q,Nyquist}}{OSR}$$

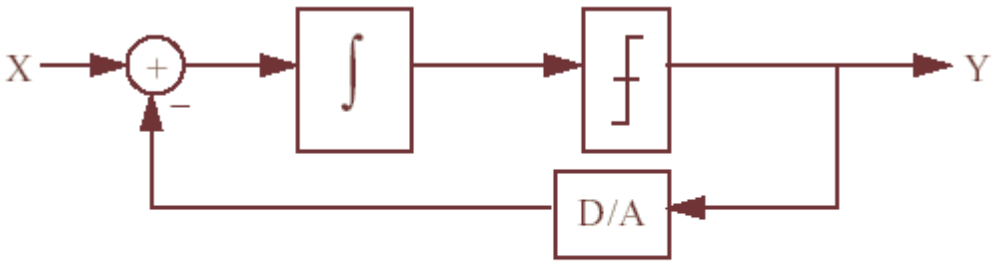
where (3.8)

$$OSR = \frac{f_s}{2 f_{BW}}$$

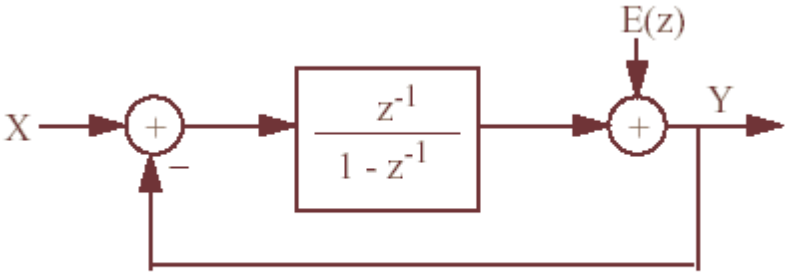
The above analysis assumes that the quantization noise spectrum is white and independent of the input signal. However, this is not true in practical implementations, but for comprehension purposes and for a simplified analysis the white noise contribution assumption is fine. A complete modeling of the quantization noise as an additive white-noise source was presented by Bennett [BEN48].

3.2.1.1 Low pass ΣΔ modulator

Sigma-delta modulators conjugate negative feedback and over-sampling to further decrease the in-band quantization noise. Figure 3.7 (a) shows a basic first-order ΣΔ modulator. Due to the negative feedback, the output **Y** will, on average, be forced to equal the input signal **X**. The input can be accurately predicted by over-sampling the input and then averaging the output, without the need for a high resolution quantizer. Figure 3.7 (b) shows a simplified linearized discrete time model of the first order modulator shown before. The equation in (3.9) shows the output **Y** as function of the input **X** and the quantization noise $\mathbf{N_q}$. In case of setting **K** equal to 1, the quantizer noise transfer function to the output has a characteristic of high pass filter and the input is only delayed, as equation (3.10) shows. The high pass characteristic of the quantizer noise transfer function attenuates the in-band quantization noise.



(a)



(b)

Figure 3.7: First order ΣΔM: (a) simplified block diagram and (b) linearized model.

$$Y(z) = \frac{z^{-1}}{1 - z^{-1}(1-K)} X(z) + \frac{1 - z^{-1}}{1 - z^{-1}(1-K)} N_q(z) \tag{3.9}$$

$$K = 1 \Rightarrow H_{signal}(z) = z^{-1} \text{ and } H_{noise}(z) = 1 - z^{-1} \tag{3.10}$$

The evaluation of the total in-band quantization noise power is made by integrating, over signal bandwidth, the noise power density weighted by the frequency response of the noise transfer function. Using the results of equations (3.8) and (3.10) and integrating them as shown by equation (3.11), an in-band noise power expression for he ΣΔ modulator of figure 3.7 is obtained. It is also shown in equation (3.11) an approximate result considering an **OSR >> 1**.

$$P_{Q,\Sigma\Delta} = \int_{-f_{BW}}^{f_{BW}} \frac{P_{Q,Nyquist}}{OSR} .(1 - z^{-1}) df \approx \frac{\Delta^2}{12} \cdot \frac{\pi^2}{3OSR^3} \tag{3.11}$$

From this result, the feedback loop in the ΣΔM works as a noise-shaping filter that tries to cancel the in-band quantization to predict the input signal. Extending this result to loop filters of higher order, one can conclude that the filters with higher order will better predict the input signal with a stronger noise shaping characteristic. Equation 3.12 extends the result to a low pass modulator of order **L** and noise transfer function equal to $(1 - z^{-1})^L$. Assuming OSR>>1 [NOR97].

$$P_{Q,\Sigma\Delta} = \int_{-f_{BW}}^{f_{BW}} \frac{P_{Q,Nyquist}}{OSR} .(1 - z^{-1})^L df \approx \frac{\Delta^2}{12} \cdot \frac{\pi^{2L}}{(2L+1)OSR^{2L+1}} \tag{3.12}$$

The dynamic range (DR) of a ΣΔM is defined according to expression (3.13), where $\mathbf{P_{X,FS}}$ and $\mathbf{P_{N,\Sigma\Delta}}$ are the full scale input signal power and the total in-band noise power of the ΣΔM, respectively.

$$DR_{\Sigma\Delta} = 10 \log\left(\frac{P_{X,FS}}{P_{N,\Sigma\Delta}}\right) \tag{3.13}$$

Assuming a single tone input signal **x[n]** with α being its normalized frequency with respect to $\mathbf{f_S}$ and its quantized amplitude equal to the modulator full scale (**FS**) with the quantization step **Δ** as defined in equation (3.6) for a quantizer with **B** bits of resolution, then the signal **x[n]** and its power $\mathbf{P_X}$ are expressed by (3.14).

$$x[n] = (2^B - 1) \cdot \frac{\Delta}{2} \cdot \sin(2\pi\alpha n) \Rightarrow P_X = \frac{\Delta^2}{8}(2^B - 1)^2 \tag{3.14}$$

Making the assumption that the quantization noise is the only noise source, the DR can be estimated by expression 3.13, using the results of equations (3.14) and (3.12). Equation (3.15) expresses the DR in this case.

$$DR_{\Sigma\Delta} = 10 \cdot \log\left[\tfrac{3}{2} \cdot (2^B - 1)^2 \cdot \tfrac{2L+1}{\pi^{2L}} \cdot OSR^{2L+1}\right] \tag{3.15}$$

From (3.15), the dynamic range can be increased by increasing the modulator order, the over-sampling ratio, or the quantizer resolution. For every doubling of the over-sampling ratio, the dynamic range increases by 3(2L+1) dB or (L+0.5) bits. It is important to point out that by increasing the order of the modulator (3rd–order or higher)

the modulator itself can become unstable or present some limit cycle oscillations problems [NOR97].

3.2.1.2 Band-pass ΣΔ modulator

Band-pass ΣΔ modulator is a close extension of the conventional low-pass modulator. The difference relies on using resonators instead of integrators in the loop filter [NOR97]. Figure 3.9 shows its basic structure. The noise-shaping characteristic of a ΣΔ band-pass modulator occurs at the frequency of the resonator. Initially, the motivation for the development of band-pass converters came from the simplicity they convey to systems that work with narrow band signals, like in RF communication systems, spectrum analyzers and some special purpose instrumentation. The employment of these modulators in early conversion to digital of RF telecom signals (at IF or RF stages) makes the system more robust, testable and suitable for multi standard operation. Digital radio telecom systems, such as cellular phones and digital audio broadcast, need advanced modulation schemes more suitable for digital implementation. The use of this class of converters makes possible the suppression of the analog IF filter of these system that have poor phase performance connected to low controllability, directly impacting in inter-symbol interference. By digitally implementing the IF filter an exactly linear phase response can obtained [NOR97].

Concerning the frequency translation approach, an image selection filter is mandatory after the mixer. Therefore, the loop filter from a continuous time band pass ΣΔ modulator could perform not only the noise shaping but also the proper image selection. This choice brings some additional advantages that are: the relaxed anti-aliasing filtering (can be very power hungry and area consuming), no need for a precision sample-and-hold circuit and relative low precision analog components, which is becoming more important as CMOS technologies scale bringing larger statistical variations [NOR97].

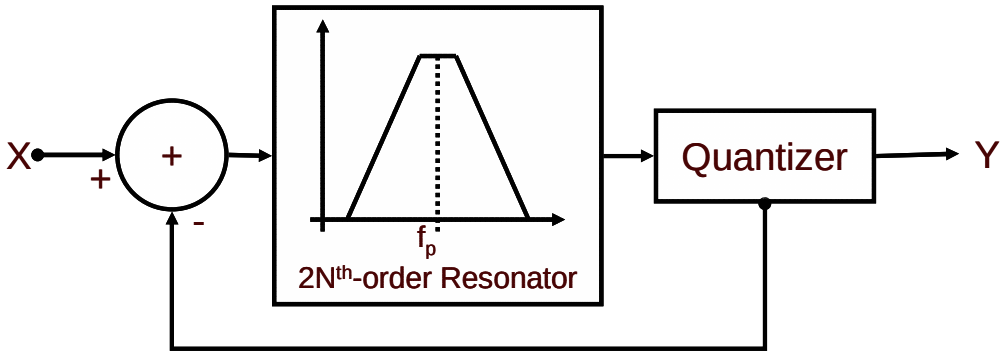


Figure 3.8: Discrete time band-pass ΣΔ modulator.

The theoretical model for this class of modulator will addressed with more details in the next chapter.

3.3 Interface Architecture

The proposed architecture interface uses the principles described in last sections to build an interface around a fairly fixed analog block to overcome the penalties that arise from reconfiguring analog parts as it in an FPAA and in an FPMA. The usage of the

frequency translation followed by the signal conversion to sigma-delta domain enlarges the frequency coverage and makes the application configurability possible at digital level with controllable losses, and capability to implement a wide spectrum of applications. The choice for moving the application configurability to digital domain comes not only from the excellent design automation tools support, but also the technology for configurable devices is well established. This approach makes possible the employment of available tools not only for digital blocks design automation but also for digital signal processing making the application design space exploration task straightforward. Moreover, the digital domain takes full advantage of Moore Law regarding scaling – power dissipation notwithstanding, certainly helped by immense gate densities - to process the input signal.

Taking into account that for most applications the input signal is band limited, one could think of reallocating this input signal to a fixed frequency band in the spectrum, so that a fixed and high performance mixed-signal section could process this translated version of the input signal. The concept of frequency translation or intermediate-frequency (IF) processing is not new. It is commonly used in telecommunication transceivers [LAT98, RAZ98] and in some instrumentation applications like chopper and lock-in phase amplifiers. However, the use of this theory for general-purpose signal processing in the context of FPMA´s is really new. This mechanism makes it possible to process signals from DC to high frequency using the same infrastructure. Making the analog section fixed, it can be optimized for high performance to fulfill constrains like signal-to-noise-ratio, dynamic range and bandwidth, for the target applications set. A configurable digital section gives the desired programmability, hopefully without analog performance penalty.

3.3.1 Analog cell structure

The analysis of all practical HF cases shows that the signal bandwidth is a small fraction of its central frequency and the carrier HF signal is the hardest limitation and not the signal bandwidth, which is a fraction of percent of the local oscillator, when considering a digitizing infrastructure. Thus, the frequency translation allows the usage of an ADC with lower sampling rate than would be required to directly digitize the original HF signal.

The treatment of narrow bandwidth signals applications have made over-sampled converters and ΣΔ modulators particularly appealing. One reason for that comes from that ΣΔ modulators trade resolution in time for resolution in amplitude, such that the use of imprecise analog circuits can be tolerated. This relaxed constraints in analog circuit blocks performance makes this technique suitable for implementation in conventional digital CMOS technologies [TAO99b, NOR97]. The use of a mixer in front of the data acquisition subsystem enables the manipulation of band limited signals located in a wide range of the spectrum, covering from DC to high frequency in a homogeneous fashion.

Using a mixer in front of the data conversion block requires a selection image filter to selects the appropriate copy of the input signal. Selecting a continuous time band pass noise shaping transfer function topology for the ΣΔ modulator, it can perform not only the noise shaping itself but also the proper image selection. This choice brings some additional advantages that are: the relaxed anti-aliasing filtering (can be very power hungry and area consuming), no need for a precision sample-and-hold circuit and

relatively low precision analog components, which is becoming more important as CMOS technologies scale bringing larger statistical variations [VAZ2003].

Analyzing the signal conditioning requirements of several applications in the field of instrumentation, control and telecommunication, the proposed interface should be able to deal with an input dynamic range (**DR**) of one or two decades. Therefore, to be able to treat such a wide dynamic range input signals, it is imperative some kind of gain control to adjust the input signal **DR** to the ΔΣ modulator input **DR**. So, the insertion of a **VGA** block between the mixer and the ΔΣ modulator could do the job of **DR** adjustments, and bringing some additional side benefits like using it for automatic gain control in some signal treatments in communication field, for example. The **VGA** block insertion in between the mixer and the ΔΣ modulator makes possible to use such a block that does not need to have the input wide frequency range response because it works in the processing frequency $\mathbf{f_p}$ range, that is fixed, and its maximum **BW** is related to maximum input **BW**. These constraints create the possibilities of designing a tuned amplifier at the processing frequency giving an additional image selection with reduction in the thermal noise power insertion [MOH2003]. The input gain control could be done in a continuous way using a digital low pass ΔΣ modulator and an analog low pass reconstruction filter.

Figure 3.9 shows the basic structure of the *fixed analog cell* (FAC). The FAC is composed by an input mixer and a continuous time (CT) N-th order band-pass ΣΔ modulator. The signals for controlling the mixer and the feedback DAC are generated by the digital reconfigurable block.

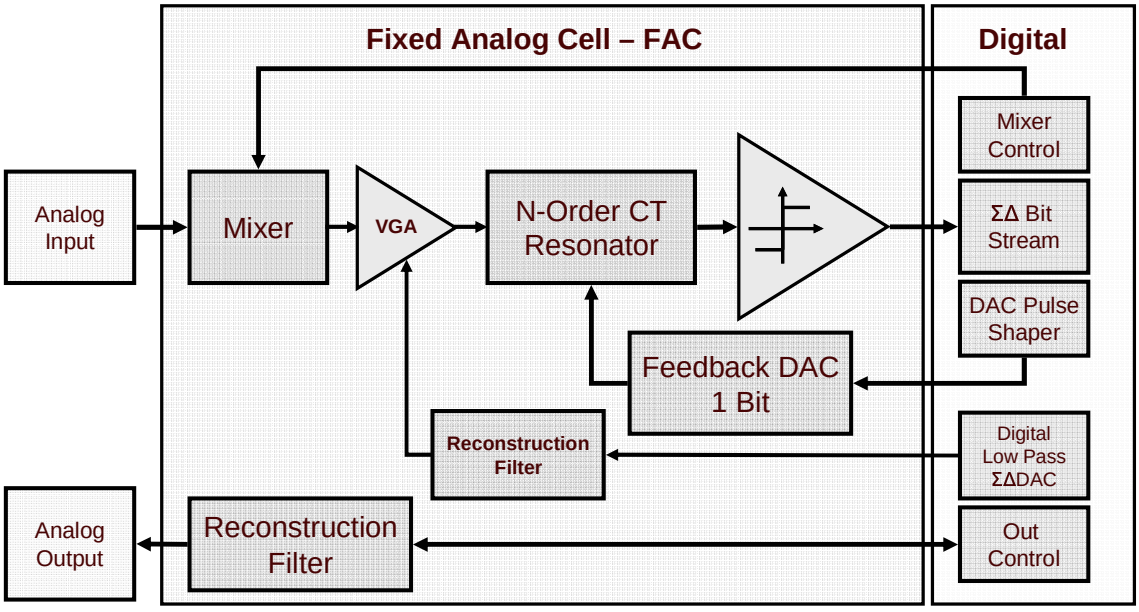


Figure 3.9: Fixed analog cell topology.

As this cell is fixed at structural and functional level, it can have an optimal design to provide the appropriate bandwidth, signal-to-noise-ratio-dynamic-range (SNRDR) and frequency coverage for the target application set. This optimization is mainly focused on the order **N** and frequency **fp** of the resonator and the mixer topology [RAZ98]. Power, area and noise budget can be used as constraints for the tailoring

process at design time. By using an intermediate frequency $\mathbf{f_p}$ (far a way from DC) to acquire the signal, instead of working at its base-band, problems like the 1/f or flicker noise, DC offset and DC offset drifts are attenuated. Moreover, the base-band signal processing mechanism allows us to process signals from DC up to HF in the same infrastructure, which would not be possible with any of the known analog or mixed-signal programmable topologies.

The channel processing flow proposed herein requires a demodulator after the input bit-stream data acquisition to bring the input signal to its base-band. The demodulator is implemented in the configurable digital block, easily. The steps that follow are associated with the application of the desired signal processing function over the input signal also using the digital infrastructure.

The desired output signal is sent back to the analog form by the output block composed by also a 1 bit DAC and a reconstruction filter [FAB2004a].

The performance modeling and high level design and constraint equations are addressed in chapter 4.

3.3.2 The programmable mixed signal interface architecture

As described earlier, the definition of the **FAC** structure opens the possibility to create a uniform mechanism for analog signal (input or output) treatment using the ΣΔ domain as working space. The single channel solution presented is extended for the multi-channel case by replicating **x** times the **FAC** according to the number of desired analog signal input channels.

Figure 3.10 shows the architecture of this general interface using a set of identical **FAC**s. Each **FAC** has it own digital block to realize the acquisition of the generated digital signal and its conversion to signal base-band for processing. It is important to remark that all signal treatment is intended to be processed in the signal base-band.

3.3.3 Discussion

In our architecture approach, the fast analog prototyping is done in a design paradigm which is altogether different. Instead of changing the circuit topology, one shifts the signal band. This has many useful implications, since the redesign or migration of the proposed configurable cell to other technologies is greatly simplified. Only the analog part of the modulator must be redesigned or targeted at the physical level to a new technology. This certainly simplifies the design process. The remaining analog processing circuits can be easily ported to a new technology, since they are simpler large signal modules, and hence consolidated digital tools are available for this.

The performance modeling and high level design and constraint equations for the proposed **FAC** are addressed in chapter 4.

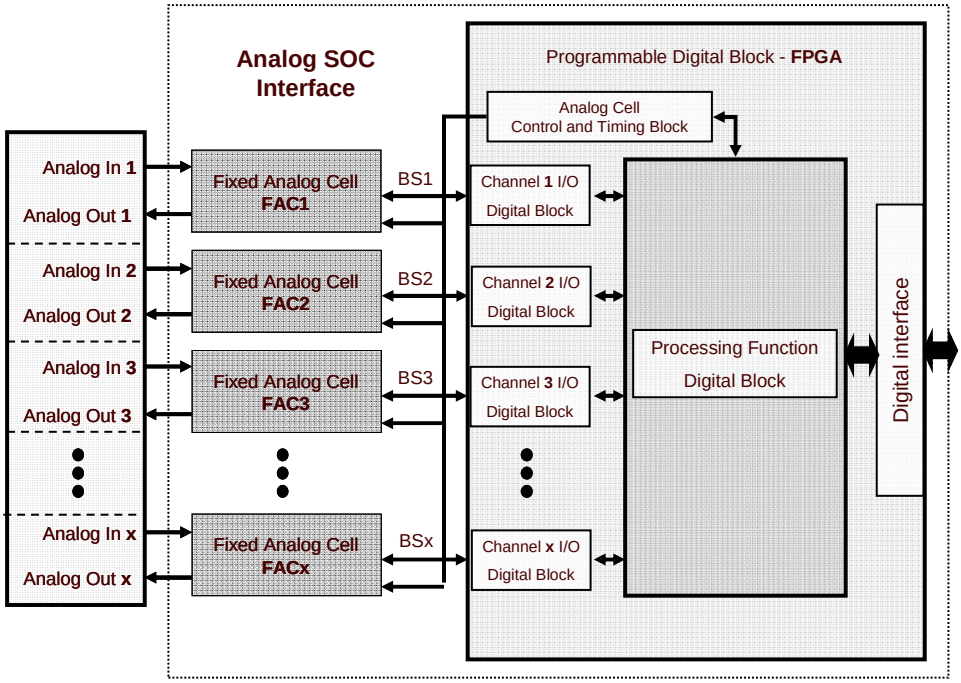


Figure 3.10: A general analog SOC interface architecture using a set of fixed analog cell as a building block.

4 THE FAC PERFORMANCE MODEL

This chapter addresses aspects regarding the performance model of the proposed general analog **SOC** interface focusing on the fixed analog cell – **FAC** construction blocks as described in last chapter. The approach is based on analyzing the **FAC** with respect to its three main blocks: the mixer, the **VGA** and the ΣΔ modulator. The performance prediction and trade-offs of each block are used to construct a behavioral model for the whole **FAC** to be used for system level design space exploration using the applications set target performance as constraint.

4.1 The Mixer Block Performance Analysis

The mixer goal is to translate an input signal from one frequency band to a different one without distorting its amplitude and/or phase. From equation 3.1, in chapter 3, the multiplication process could do this job. Equation 3.2 also shows that the multiplication process implies that the resulting translated signal amplitude depends not only on the amplitude of input signal $\boldsymbol{A_i}$ but also from local oscillator amplitude $\boldsymbol{A_{LO}}$. Therefore, the local oscillator amplitude must be precisely controlled to avoid distortion on output signal. Indeed, the wish is the dependency of mixer output signal amplitude only on the amplitude of input signal. This way, the mixer output signal could be expressed by

$$v_P(t) = A_i \cos(\omega_i t + \Phi_i) \times \mathrm{sgn}(LO)\,. \tag{4.1}$$

Equation 4.1 conceptually says that the input should be multiplied only by the sign of the local oscillator signal what mathematically is a square wave (1) multiplication where its frequency is equal to the local oscillator frequency as shown by 4.2.

$$v_P(t) = A_i \cos(\omega_i t + \Phi_i) \times square(\omega_{LO} \cdot t) \tag{4.2}$$

Expanding expression 4.2 using the Fourier series of a square wave with amplitude **1**, we obtain equation 4.3.

$$\begin{aligned} v_P(t) &= A_i \cos(\omega_i t + \Phi_i) \times \frac{4}{\pi}\left[\cos(\omega_{LO} t) + \frac{1}{3}\cos(3\cdot\omega_{LO} t) + \cdots + \frac{1}{2n-1}\cos((2n-1)\cdot\omega_{LO} t) + \cdots\right] \\ v_P(t) &= \frac{2A_i}{\pi}\left[\cos(\omega_P t) + \cos[(\omega_P + 2\omega_{LO})t]\right] + \frac{2A_i}{3\pi}\left[\cos[(\omega_P - 2\omega_{LO})t] + \cos[(\omega_P + 4\omega_{LO})t]\right] + \cdots \\ v_P(t) &= \frac{2A_i}{\pi}\cos(\omega_P t) + \cdots \end{aligned} \tag{4.3}$$

It is clear from 4.3 that the mixing process generates high order harmonics copies of the input signal at the mixer output spectrum that must be filtered out by the image

rejection filter or channel selection filter. Our interest is only on the copy created at ω_p (processing frequency band), because it is in that position that the input signal is going to be acquired.

Figure 4.1 illustrates the mixing process using switching as described above. In this example, the processing frequency band (**PF**) is centered at 10 MHz and the input signal is at 11.5 MHz, so one possible frequency choice (lowest) for the **LO** is 1.5 MHz to translate the input to the processing band.

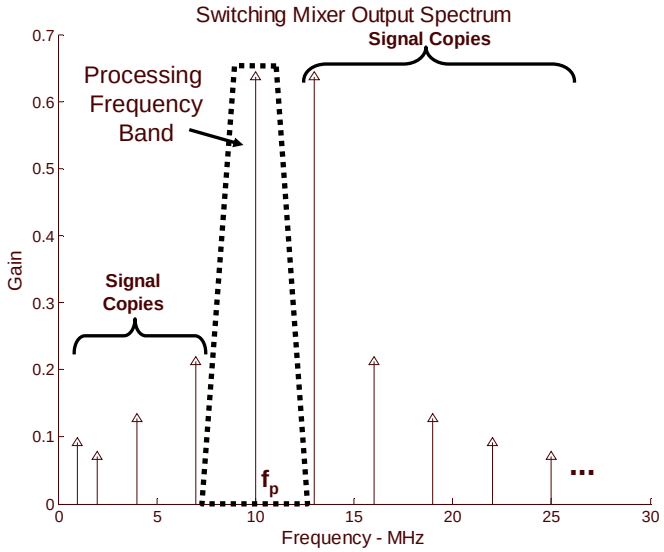


Figure 4.1: Mixer output spectrum for a **LO** frequency equal to 1.5 MHz and an input signal at 11.5 MHz, the processing frequency is 10 MHz.

A mixer is inherently a non-linear block because it creates frequencies at its output that do not exist at the inputs. However the multiplier block could be substituted by a multiplexer that commutes between a positive and negative gain once each **LO** half period cycle, as shown in figure 4.2. As a result, for each half period of **LO** the transfer function from the input to the output **PF** is linear. This way, the mixer can be considered as a linear-time-variant system whose its gain changes cyclically according to the **LO** signal.

At this point, it is important to define the performance parameters that characterized a mixer topology. The performance metrics are the conversion gain, bandwidth, 1 dB compression point, third order intercept point (**IP3**), noise figure and port-to-port isolation [TOU2002].

4.1.1 Performance metrics of mixers

It is not the goal of this work to deep into details on the description of the metrics commonly used to evaluate performance of mixers. We are going only to give a brief review of the basic parameters and how they are related to this work.

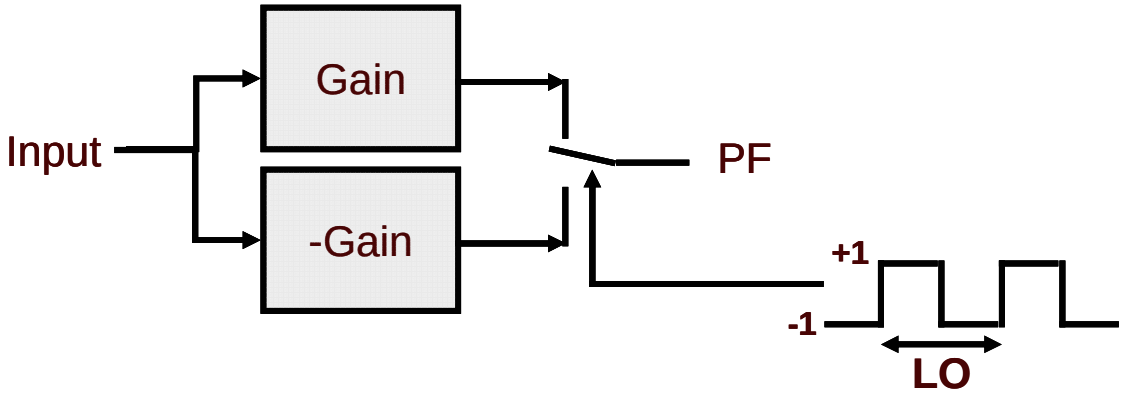


Figure 4.2: A linear time variant model for a switching mixer.

4.1.1.1 Conversion gain and bandwidth

The conversion gain and bandwidth definitions are similar to the ones used to characterize linear systems with the distinction that the input and output frequency are in different spectral positions.

The definition of conversion gain is the ratio between the signal at input port (**RF** port, for example) and the signal at **PF** port (**IF** port, in general **RF** mixers). For example, in figure 4.2 the conversion gain is around 0.64. In most mixers (active), the conversion gain drops when the **LO** frequency gets higher what sets some sort of "bandwidth".

From the fact that mixers have two distinct operating frequencies (input and **FP**) so it is possible to define two different bandwidths. The input bandwidth defines the frequency coverage on the input port. On the other side, the output bandwidth is the one seen by the output signal [TOU2002, LEE98b].

4.1.1.2 1dB compression point (P_{1dB}) and third order intercept point (IP3)

The linearity of a mixer is characterized by 1dB compression point (**P_{1dB}**) and third order intercept point (**IP3**) [TOU2002, LEE98b].

The 1dB compression point (**P_{1dB}**) measure the deviation from the ideal linear relationship between the input and output, actually it identifies the input mixer power at which the conversion gain drops 1dB, as shown in figure 4.3.

The other parameter called third order intercept point (**IP3**) gives information about the potential **SNR** or **SFDR** degradation in the case a interferer signal that is close to the desired input tone, it is also identified in figure 4.3.

4.1.1.3 Port-to-port isolation

Another parameter that characterizes the dynamic behavior of a mixer is the port-to-port isolation that is especially important for high frequency operating systems. It inform about the leakage of either the **LO** signal to input and output ports. The **LO** leakage to input port can cause self-mixing leading to **DC** offsets that could corrupt the output. The **LO** and input to **FP** port leakage is not a big problem when the **LO** and input signal are located outside of **FP** band that actually occurs in most applications.

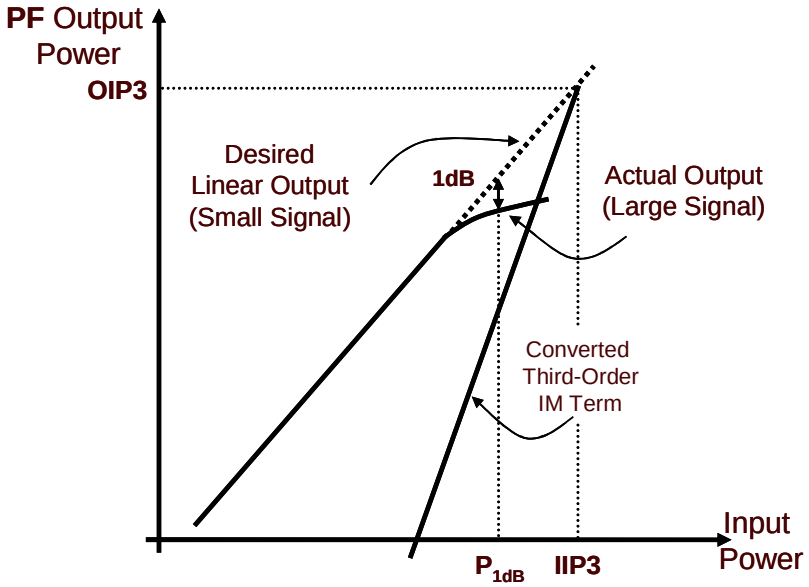


Figure 4.3: The graphical representation of the 1dB compression point ($\mathbf{P_{1dB}}$) and third order intercept point (**IP3**).

4.1.1.4 Noise figure and noise factor

The classical definition of noise factor (**F**) is the **SNR** degradation when a signal passes through a circuit and the noise figure (NF) is the noise factor expressed in decibels, as expression 4.4 shows.

$$F \equiv \frac{\text{total output noise power}}{\text{output noise power due to input source}} \qquad (4.4)$$
$$NF \equiv 10\log F$$

The determination of the noise figure must be carefully planed, because actually there is two possible input frequencies that can generate the PF signal. One is the desired input and the other is an image, these two signals are called sidebands and they are $2\omega_{IF}$ apart from each other. So, when computing the NF, for example, it is necessary to state if the result is for *double-sideband* (**DSB**) or *single-sideband* (**SSB**). The **NF SSB** is 3 dB higher than **DSB NF**, assuming the conversion gain is the same for both frequencies.

4.1.2 Mixer topology

There are several architectures to implement a down or up conversion mixer as shown in [RAZ98, LEE98b]. Looking at the **CMOS** technology components naturally incorporate excellent switches, so these switches could be used to develop high performance multipliers employing switching techniques. This approach has not only a great appealing by its simplicity but also it require no biasing making it a choice for very low power applications [LEE98b].

A double-balanced passive mixer simplified structure is shown by figure 4.4. Indeed, this topology is a bridge of four **CMOS** transistors driven by a **LO** with two counter-phase, in each clock phase only two switches are on in opposite branches creating a balanced output. The simplest architecture is based on a set of CMOS transistors working as switches (transmission gates), and this realization is called

passive mixer. The multiplication process involved in this topology is exactly the same developed early in this chapter, so all harmonic copies of the input are created as figure 4.1. Actually, if the **LO** clocking signals do not have an exact 50% duty it also creates copies of the input related even harmonics of the clocking signal. Those harmonic input copies are also suppressed by the channel selection filter that follows.

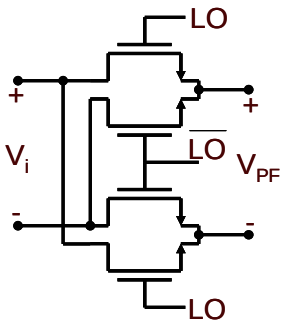


Figure 4.4: A four transistors simplified double-balanced passive mixer topology.

This mixer topology has two main advantages. First, it achieves a higher IP3 (third harmonic intercept point or inter-modulation product), since the transistors are submitted to a large $\mathbf{v_{gs}}$ overdrive. Second, there is no power consumption from the power supplies. Moreover, it allows easy generation of the local oscillator signal, as it can be a square wave.

Of course, it has some disadvantages. One of this drawbacks could be conversion gain (equation 4.3) less than one ($2/\pi \rightarrow$ -4 dB), that could be compensated by the $\Sigma\Delta$ modulator input scaling or a **VGA** block.

4.1.3 Passive mixer dynamics and performance analysis

In last section the mixer topology that is going to be used in the proposed **FAC** was presented. Now, we are going to proceed in the mixer performance estimation dividing it into two parts: one regarding the **LO** dynamics and other regarding flicker noise and distortion.

4.1.3.1 Noise and distortion

Exist a strong dependency of the noise figure and **IP3** (distortion) from the **LO** driving waveform, since both parameters are closely related to the transistor channel resistance at “on” state [LEE98b].

The “on” channel resistance impacts on mixer thermal power noise injection, so to keep it at low levels it must be small valued that implies large area transistor. The usage of large transistor on the other hand brings not only an increase in the gate capacitance as also an increment in the gate-drain and gate-source parasitic capacitance. Higher gate capacitances has impact on power consumption and driving capabilities of **LO** do to load increasing and possible reducing the upper limit on frequency coverage. A side effect is the increment in **HF** noise injection do to higher currents switching. The raise in the overlap capacitances reduces the port-to-port isolation.

To obtain a high **IIP3**, it necessary to maintain the “on” state channel resistance as constant as possible what requires large $\mathbf{v_{gs}}$ overdrive voltage.

4.1.3.2 The LO dynamics performance impact

The mixer control signals are derived from a **LO** timing system that indeed is a frequency synthesizer comprising a **CMOS** oscillator or **VCO** / **PLL** circuit or a combination of them [LEE98b]. The **LO** circuitry, as all circuits, are subject to noise sources. A question rises up: what is it the impact of the quality of **LO** signals generation on the mixer performance? Regarding the fact that those control signal have digital format, noise superimpose on digital signals, with amplitude below the noise margins of the mixer drivers (digital buffers), will not impact on the mixer noise performance [MAR2000]. What is really import is sharp transition and regularly spaced, so time uncertainties in transition edges can cause mixer performance degradation and intuitively its impact increases as input signal frequencies get higher. This way, one of the most important aspects is the time jitter at the mixer switching control signal. The presence of time jitter in the transition edges of those signal, as figure 4.5 shows, impacts the **SNR** at the mixer output with strong dependency on the switching frequency [SHI90]. Actually, the time jitter in the control signal will be one the most limiting factors for high frequency operation due to **SNR** degradation, so it is important to model this effect.

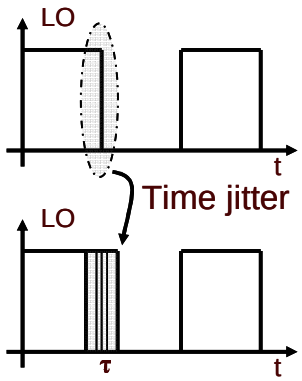


Figure 4.5: The time jitter on mixer control signal edges.

From equation 4.2, the mixer control signal is a square wave, whose frequency is equal to ω_{LO}, a Fourier series can be used to express it in terms of its harmonics as shown in expression (4.3). The presence of time jitter in the switching signal represents an uncertainty in the switching time and can be modeled as shown in equation (4.5), where τ represents the time jitter.

$$t_j = t + \tau \tag{4.5}$$

Taking an input tone with angular frequency ω_i and amplitude **A**, the resulting image at ω_P in presence of time jitter is given by (4.6).

$$v_P = \frac{2A}{\pi}\left[\cos(\omega_{LO}\tau)\cos(\omega_P t) + \sin(\omega_{LO}\tau)\sin(\omega_P t)\right] \tag{4.6}$$

In equation (4.6), there are two terms at the desired image both being modulated by the time jitter (one in terms of *cosine* and the other in terms of *sine* functions). Considering **RMS** value of time jitter τ a small fraction of ω_{LO}, so the product $\tau\omega_{LO}$ is much less than 1, making possible to approximate (4.6) by

$$v_P \approx \frac{2A}{\pi}\left[\cos(\omega_P t) + \omega_{LO}\tau \sin(\omega_P t)\right] \qquad (4.7).$$

From the last equation, the signal (first term) and the noise (second term) contribution are separated, making possible to calculate the **SNR** at the output of the mixer with a time jitter distribution with zero mean and **RMS** value of $\boldsymbol{\tau_{RMS}}$ [SHI90].

$$SNR_{Mj} = 10\log\left(\frac{P_{sig}}{P_{Mj}}\right) = 10\log\left(\frac{A^2/2}{(\omega_{LO}\cdot\tau\cdot A)^2/2}\right) = -20\log(2\pi f_{LO}\tau_{RMS}) \qquad (4.8)$$

Using (4.8) it is possible to estimate the signal-to-noise ratio degradation due to the presence of time jitter at the mixer control signal, that it is one of the most limiting factors for achieving high frequency of operation for the proposed interface. The mixer thermal noise comes from the transistor channel at the "on" state of the switches due to the fact they operates in triode region that can be done reasonably low in the range of few Ohms.

Till this point, we have modeled the time jitter as randomly phase shift. Another question rises up: what really happens with this phase noise and its impact in the spectrum when using practical circuits **VCO** and **CMOS** oscillators as timing sources for the mixer? The first approach is to think that edge transition time occurs at each half period $\mathbf{T_{LO}}/2$ plus an *identically and independently distributed* time uncertainty $\boldsymbol{\Delta_k}$, so this time is

$$t_k = k\frac{T_{LO}}{2} + \Delta_k \ . \qquad (4.9)$$

In practical VCO and oscillators, the simple transition time model given by (4.9) does not effectively represents what really happens in those timing systems. Actually, the time jitter $\boldsymbol{\Delta_k}$ is accumulated from one transition to another as equation (4.10) shows [CHE99, HAJ98, RAZ96].

$$t_k = k\frac{T_{LO}}{2} + \sum_{n=1}^{k}\Delta_n \qquad (4.10)$$

To illustrate the difference between the two approaches, we built a mixer model to evaluate the impact them at mixers dynamics performance using the Matlab® as platform. A Gaussian random time noise generator with zero mean and standard deviation normalized to the **LO** period $\mathbf{T_{LO}}$ is used to generate the time uncertainties to create the transition edges of the mixer driving square wave, as defined in equation 4.2, using the transition time definitions giving by 4.9 and 4.10. It is interesting to refresh that the power spectrum density (**PSD**) of a Gaussian distribution is white and, indeed, from equation 4.6, the effect of time jitter in the processing frequency appears as an additional modulating source. So we should expect that applying 4.9 rule a white noise floor being added to the desired input signal, as figure 4.6 shows. On the other hand, using 4.10 that it integrates the time jitter over the time a skirt shape around the input signal shows up, resembling the same behavior of the **1/f** noise near **DC**. The "skirt" effect comes from the integration carried out in time that it promotes a $\mathbf{1/f^2}$ noise distribution **PSD** reshape, this effect is also reported and related to phase noise in oscillators [TAO99a, HAJ98, RAZ96, CHE99]. In figure 4.6, the time jitter used in the simulation identified by '**o**' and '□' were the form expressed by equation 4.10 using a

Gaussian noise source with **RMS** value equal to the local oscillator signal period ($\mathbf{T_{LO}}$) divided by 10000 and 100, respectively; the '*****' simulation used a non-accumulated Gaussian noise source with **RMS** value equal to the local oscillator signal period ($\mathbf{T_{LO}}$) divided by 10000.

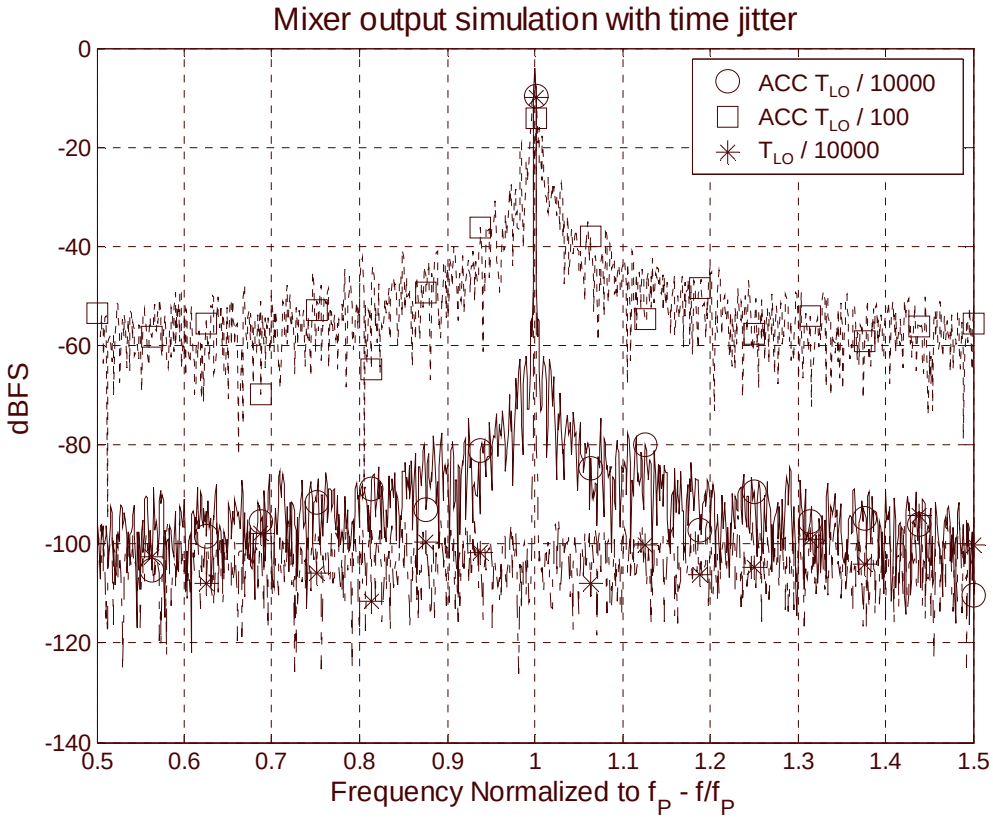


Figure 4.6: Influence of time jitter in the control signals of a mixer using accumulated and non-accumulated time uncertainties.

Another aspect concerning the dynamic behavior of the mixer block is frequency dynamic range necessary to cover desired input bandwidth. Taking into account that this interface is going to process signals from **DC** to $\mathbf{f_{max}}$, the *frequency dynamic range* – **FDR** of the mixer input and mixer control unit is defined by expression (4.11).

$$FDR = \frac{f_{\max} + f_p}{f_p} = \frac{f_{\max}}{f_p} + 1 \qquad (4.11)$$

From (4.11), it is clear that the processing frequency $\mathbf{f_p}$ location directly impacts the frequency dynamic range of the timing generation unit.

The next step is to model the performance of the ΔΣ modulator block.

4.2 The CTBPΔΣ Modulator Block Performance Analysis

Low-pass (**LP**) and band-pass (**BP**) continuous time ΔΣ modulators have increasing use in high frequency (**HF**) applications as basic building blocks for data conversion in telecommunication arena [SUS2004]. They are particularly attractive because their robustness and tolerance to low quality components that make them particularly

interesting to construct **HF** (tens of MHz) data conversion interfaces with relatively high resolution (**SNDR** around 70 dB) employing low resolution **AD** and **DA** converters.

In the following, we will discuss the performance metrics of ΔΣ modulators and their impact on the **FAC** noise model.

4.2.1 Performance metrics of ΔΣ modulators

As it was stated earlier, the ΔΣ modulator low resolution quantizer is inside a feedback loop that moves away the excess quantization noise from signal band, the resultant shaped noise frequency profile is set by the noise shaping loop filter. Actually, the ΔΣ modulator output is a signal quantized stream buried in quantization noise and distortion (due to non-linear quantization process). The quantizer output stream can be either a bit or a word stream depending on the number of quantization level use in the quantizer block. The signal bandwidth **SBW** of interest is related to sampling frequency through the **OSR**, as equation 4.12 shows. This way, the overall performance of the ΔΣ modulator is best evaluated by its **SNDR** versus input signal power; a typical performance curve is depicted in figure 4.7. The **SNDR** expresses the ratio between the signal power over the total noise and distortion powers inside the signal bandwidth, so making possible a straightforward determination of an *equivalent number of bits* – **ENoB** through the classical equation 4.13.

$$SBW = \frac{f_s}{2OSR} \tag{4.12}$$

$$ENoB = \frac{SNDR_{dB} - 1.76}{6.02} \text{ bits} \tag{4.13}$$

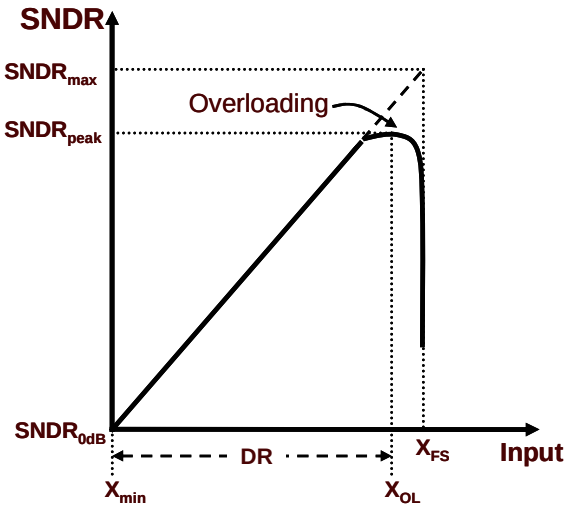


Figure 4.7: Typical **SNR** characteristic of a ΔΣ modulator

The minimum input signal power $\mathbf{X_{min}}$ is the 0 dB **SNDR** point where the input and noise power have the same magnitude. From this $\mathbf{X_{min}}$ point till the overloading point $\mathbf{X_{OL}}$, the **SNDR** increases as input increases; this part of ΔΣ modulator response curve is dominated by the signal power. The overloading point is characterized by a flattening in the **SNDR** that is caused by a high growth in quantization error power owing to quantizer overload, the peak **SNRD** is achieved in this region. An increase in the input

power ahead the overloading point augments the probability of reaching instability in the ΔΣ loop getting a sharp drop in the **SNRD**. The input *dynamic range* – **DR** is defined as the ratio between the input overloading point ($\mathbf{X_{OL}}$) to the minimum input power ($\mathbf{X_{min}}$). Figure 4.7 shows those points of interest.

$$DR = 10\log\frac{X_{OL}}{X_{\min}} \tag{4.14}$$

In some applications in telecommunication field, for example at maximum input signal, the ratio between the maximum signal power and the highest harmonic peak in signal band is an important parameter and is called *spurious free dynamic range* – **SFDR** (equation 4.15 and figure 4.8).

$$SFDR = 10\log\frac{P_{sig}}{P_{spurious}} \tag{4.15}$$

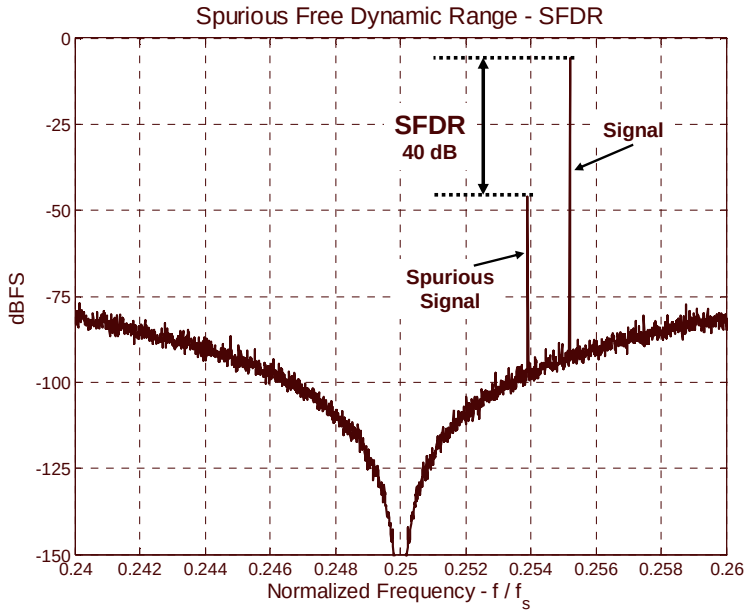


Figure 4.8: The **SFDR** definition.

4.2.2 CTBPΔΣ Modulator topology

For developing the **CTBPΔΣ** modeling, it is necessary to define the modulator *noise transfer function* – **NTF** definition, since the modulator whole performance is closely linked to the **NTF** shape. During the **NTF** definition phase, the designer must keep in mind that the ΔΣ modulator is a feedback system and as a feedback system instability problems can show up since the **NTF** not only impacts the quantization noise shaping but also the ΔΣ loop stability. So, we chose as starting point a discrete time low pass ΔΣ modulator – **LPΔΣMod** prototype [NOR97, VAZ2003] where loop stability and noise shaping are well defined. Figure 4.9 shows a recursive way of building an order **N LPΔΣMod** (**LPΔΣModN)** from a first order modulator (**LPΔΣMod1**). The **LPΔΣModN NTF** and the *signal transfer function* – **SFT** obtained by this method are given by equations 4.16 and 4.17 respectively.

$$NTF(z) = (1 - z^{-1})^{N} \tag{4.16}$$

$$STF(z) = z^{-1} \tag{4.17}$$

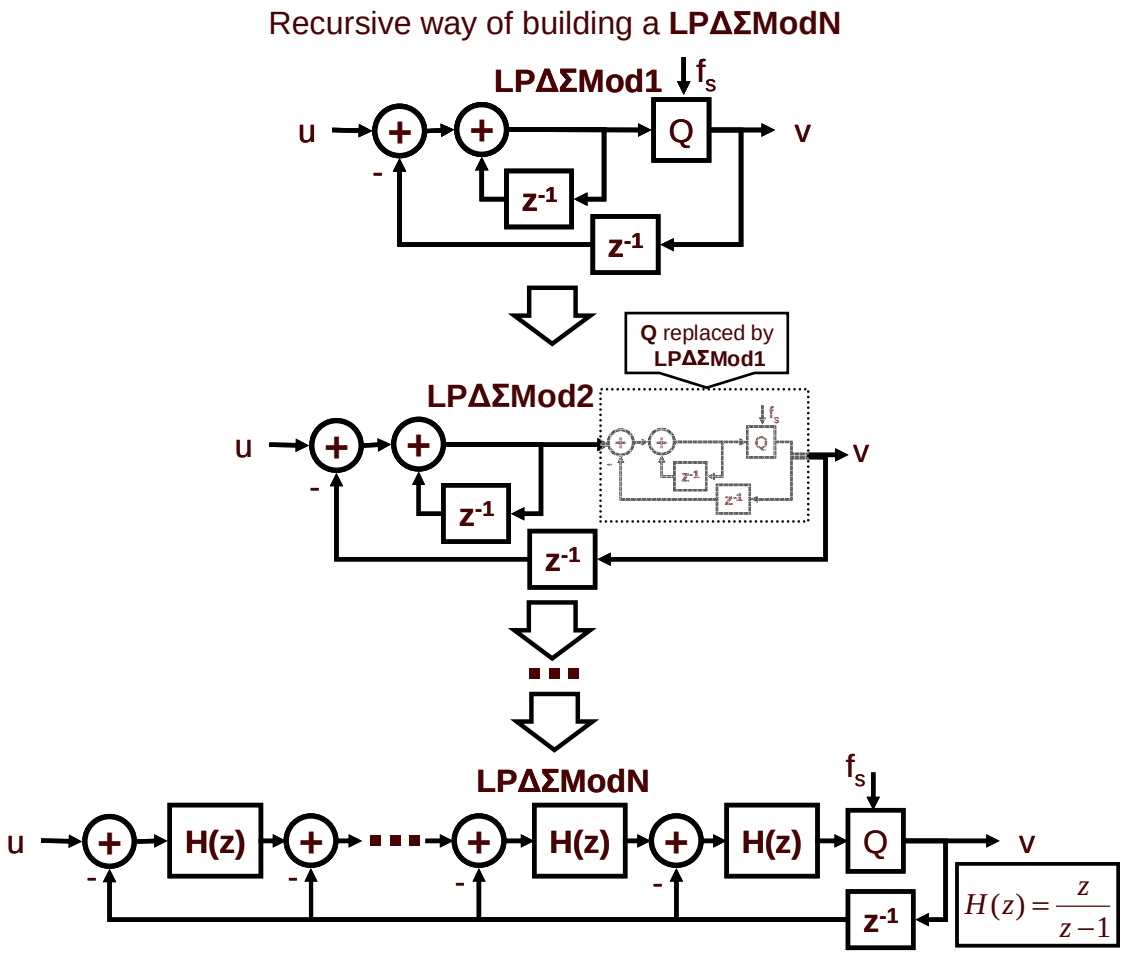


Figure 4.9: A recursive way of obtaining a **LPΔΣModN** from a **LPΔΣMod1**.

A commonly used in filter design low-pass to band-pass frequency transformation is used to transform the low-pass prototype described above into a discrete time band-pass ΔΣ modulator prototype. Equation 4.18 shows the low-pass to band-pass frequency transformation [NOR97].

$$z_{LP} = -z_{BP} \cdot \frac{z_{BP} - \cos(\Omega_P)}{1 - \cos(\Omega_P) \cdot z_{BP}}$$
$$\Omega_P = 2\pi \frac{f_P}{f_s} \tag{4.18}$$

Applying the transformation rule given by 4.18 in equations 4.16 and 4.17, it results in the band-pass prototype version of the original low pass, as shows equations 4.19 and 4.20. From 4.19 and 4.20, it is noted that the final order is twice the order of the low pass version.

$$NTF_{BP}(z) = \left(\frac{1 - 2 \cdot \cos(\Omega_P) z^{-1} + z^{-2}}{1 - \cos(\Omega_P) z^{-1}} \right)^N \tag{4.19}$$

$$STF_{BP}(z) = z^{-1} \cdot \frac{z^{-1} - \cos(\Omega_P)}{1 - \cos(\Omega_P) \cdot z^{-1}} \tag{4.20}$$

Figure 4.10 shows the band-pass general architecture drawn from the general **LPΔΣModN** shown in figure 4.9. Now, the **N** integrators are replaced by **N** resonator at the target band pass frequency.

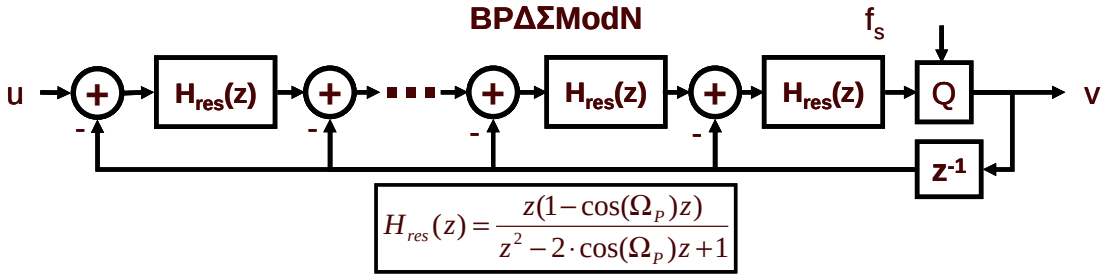


Figure 4.10: General topology of a **BPΔΣModN**.

It is important to observe that the transformation described earlier maintains the performance and stability characteristics of low-pass prototype used as starting point. Figures 4.11 and 4.12 show the simulation results of four different ΔΣ modulators; one **LPΔΣMod2** and three **BPΔΣMod2**. The **LP** was used as a prototype to generate three different **BPΔΣMod2** at frequencies **1/16**, **1/8** and **1/4** of $\mathbf{f}_s$. The four simulations show a close relationship between the **LP** prototype noise shaping and **SNDR** with the **BP** versions.

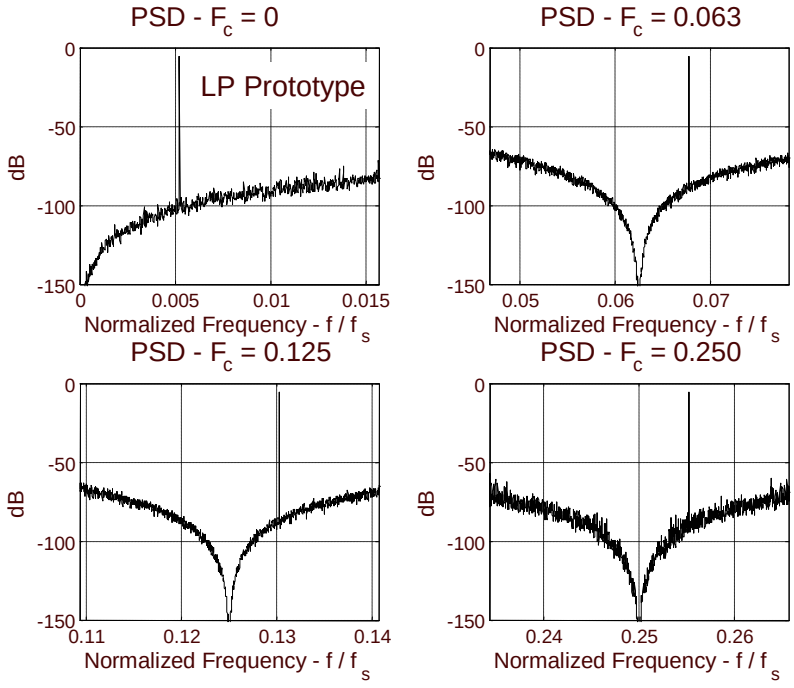


Figure 4.11: The **PSD** simulation results of a second order **LP** prototype and three different **BP** ΔΣ modulators obtained through **LP** to **BP** frequency transformation.

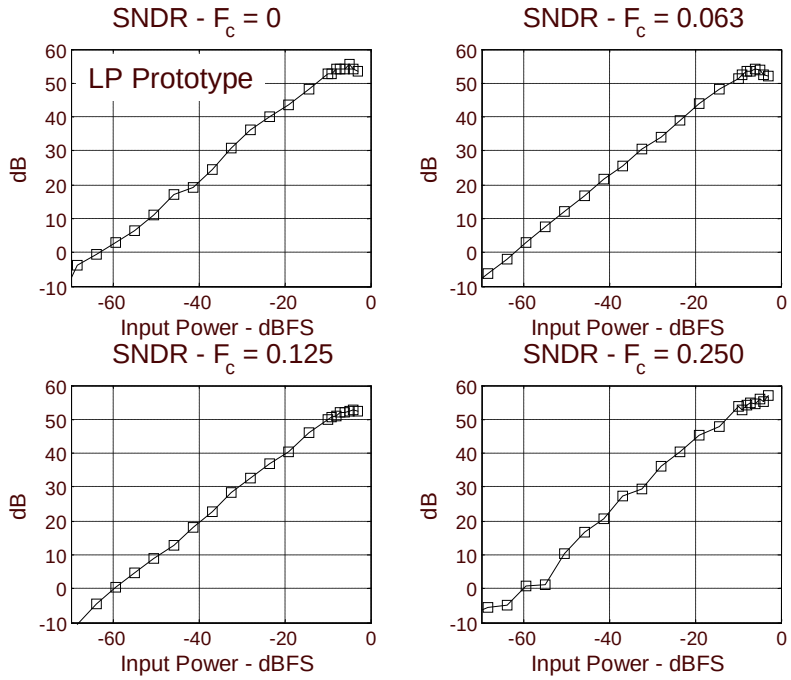


Figure 4.12: The **SNDR** simulation results of a second order **LP** prototype and three different **BP** ΔΣ modulators obtained through **LP** to **BP** frequency transformation.

The next step is a continuous time equivalent BPΣΔ modulator design using some discrete to continuous time transformations, like the modified **Z** transform or pulse invariant transform. The objective in this procedure is to find a continuous time transfer function that mimics the prototype discrete time behavior at each sampling instant [NOR97]. Expression (4.21) shows this general procedure, where **T** is the sampling period. $\mathbf{H_{Res}}(z)$ is the discrete time resonator transfer function designed according to equation (4.19), which fulfills the target performance specifications. The right hand side of equation 4.21 is a composition of $\mathbf{R_{DAC}}(s)$ and $\mathbf{H_{CTR}}(s)$ that are the **DAC** and the continuous time equivalent resonator transfer functions, respectively. The $\mathbf{R_{DAC}}(s)$ transfer function incorporates the **DAC** pulse shape what makes possible the more convenient **DAC** pulse waveform selection in the synthesis process for the set of applications.

$$Z^{-1}\left\{H_R(z)\right\}_{t=nT} = L^{-1}\left\{R_{DAC}(s)H_{CTR}(s)\right\}_{t=nT} \qquad (4.21)$$

The parametric modeling of the band-pass ΔΣ modulator is going to be presented in next section.

4.2.3 CTBPΔΣ Modulator performance analysis and dynamics

The starting point for modeling the **BPCTΔΣ** modulator block is a discrete time prototype, as described in last section. The recursive topology presented in figure 4.10 can be modified using classical block manipulation to have only one feedback path, as figure 4.13 shows. So, the discrete time N^{th}-order resonator transfer function having all poles at the same place in the unit circle can be written as

$$H_R(z) = \left[\frac{z\cdot(1-\cos(2\pi k)\cdot z)}{z^2-2\cos(2\pi k)z+1}\right]^N$$
$$k \equiv \frac{f_p}{f_s} \qquad (4.22)$$

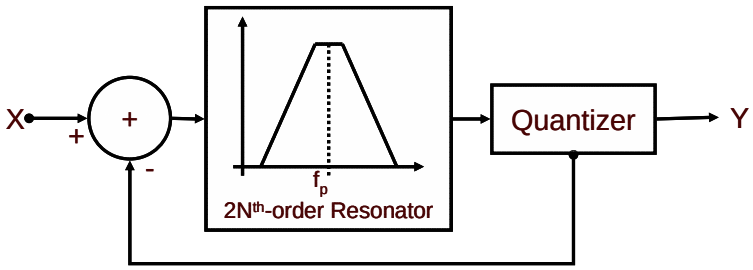


Figure 4.13: Discrete time band-pass ΣΔ modulator with single feedback path.

Using the conventional linear model for the ΣΔ modulator where the quantization error is assumed as a white noise source independent of the input signal [NOR97], the signal and noise transfer function can be rewritten as shown in (4.23) and (4.24), respectively, with $\mathbf{N_R}(z)$ given by 4.25.

$$STF(z) = \frac{(N_R(z))^N}{(1-2\cos(2\pi k)z^{-1}+z^{-2})^N+(N_R(z))^N} \qquad (4.23)$$

$$NTF(z) = \frac{(1-2\cos(2\pi k)z^{-1}+z^{-2})^N}{(1-2\cos(2\pi k)z^{-1}+z^{-2})^N+(N_R(z))^N} \qquad (4.24)$$

$$N_R(z) = 1-\cos(2\pi k)z^{-1} \qquad (4.25)$$

The noise power spectral density (**PSD**) shaped by the loop filter is expressed by equation (4.26), where **Δ** is the quantization step from a **B** bits quantizer, $\mathbf{f_s}$ is the sampling frequency and $\mathbf{Q_{FS}}$ is the quantizer full scale range.

$$N_{PSD}(\omega) = \frac{\Delta}{12\cdot 2\pi f_s}\left|NTF(\omega)\right|^2$$
$$\Delta \equiv \frac{Q_{FS}}{2^B-1} \qquad (4.26)$$

Integrating expression (4.26) over the signal bandwidth **BW** as (4.27) states we obtain the total quantization noise power $\mathbf{P_{QN}}$ into the signal band.

$$P_{QN} = \int_{\Omega_P-\Omega_{BW}/2}^{\Omega_P+\Omega_{BW}/2} N_{PSD}(\Omega)d\Omega$$
$$\text{with } \Omega_{BW} \equiv \frac{\pi}{OSR} \text{ and } \Omega \equiv \frac{2\pi\cdot f}{f_S} \qquad (4.27)$$

The Taylor series expansion of 4.24 around the center frequency $\mathbf{f_p}$ assuming that **BW** << $\mathbf{f_p}$, as equation 4.28 shows, is used to calculate the result of equation 4.27.

$$N_{PSD}(\Omega) = P_Q \left| NTF(e^{j\Omega}) \right|^2 = P_Q \left[A_0 + A_1\delta + A_2 \cdot \delta^2 + A_3 \cdot \delta^3 + O_3(\delta) \right]^N$$

$$\delta \equiv \Omega - \Omega_P \tag{4.28}$$

$$A_0 = 0, \quad A_1 = 0, \quad A_2 = 4 \quad \text{and} \quad A_3 = \frac{4}{\tan(\Omega_P)}$$

The values of the Taylor's series coefficients $\mathbf{A_0}$, $\mathbf{A_1}$ and $\mathbf{A_2}$ actually reflect how the **NTF** behaves in the vicinity of Ω_P, it has a local minimum at this point with gain equal to zero that was forced by design. The third order effect $\mathbf{A_3}$ shows a small dependency on the processing relative position in the spectrum, that in fact we forced this also by starting with a **LP** prototype and remapping it to another center frequency Ω_P away from **DC**, that was already shown in figure 4.11. In fact, the relative position of Ω_P does not impact directly in inherent quantization noise performance and its selection is mainly application driven, that it will be discussed this later on. Therefore, truncation 4.28 in its second order term and applying 4.27 we obtain expression 4.29 that predicts the total quantization noise power in the signal band. The term $\mathbf{Q_{FS}}$ is the quantizer full scale range.

$$P_{QN} \approx \frac{\pi^{2N} {Q_{FS}}^2}{12(2^B - 1)^2 (2N+1) OSR^{2N+1}} \tag{4.29}$$

The modulator **SNR** is obtained by taking the input signal power and dividing by the quantization noise power $\mathbf{P_{QN}}$, so considering an input tone signal with amplitude **A** inside the signal band, i.e., near $\mathbf{f_p}$, the ΣΔ modulator **SNR** is estimated by expression 4.30. In figure 4.14 and 4.15, the simulated and predicted **SNR** are shown for a second order (**N=2**) BPΔΣMod employing two different quantizers one single bit quantizer (**b=1**) and one four bit quantizer (**b=4**), respectively, for two different **OSR**.

$$SNR_{BP\Sigma\Delta M} = \frac{A^2/2}{P_{QN}} = \frac{6 \cdot (2^B - 1)^2 \cdot (2N+1) \cdot OSR^{2N+1}}{\pi^{2N}} \frac{A^2}{{Q_{FS}}^2} \tag{4.30}$$

From expression 4.30, the output dynamic range of the modulator is approximately predicted by using the maximum allowable input for which the ΔΣ remains stable, this value is around half the quantizer full scale [NOR97], so the **DR** is given by equation 4.31.

$$DR_{BP\Sigma\Delta M} = \frac{{Q_{FS}}^2/8}{P_{QN}} = 10 \cdot \log\left[\frac{3 \cdot (2^B - 1)^2 \cdot (2N+1) \cdot OSR^{2N+1}}{2\pi^{2N}} \right] \text{ in dB} \tag{4.31}$$

Expression 4.30 synthesizes all characteristic parameters for the band-pass ΣΔ modulator design and can be used for design space exploration for the determination of the most appropriate value for **OSR**, modulator order **N** and the quantizer number of bits **B**. After the definition of these parameters, we can go ahead and find an equivalent continuous time BPΣΔ modulator as mentioned in last section.

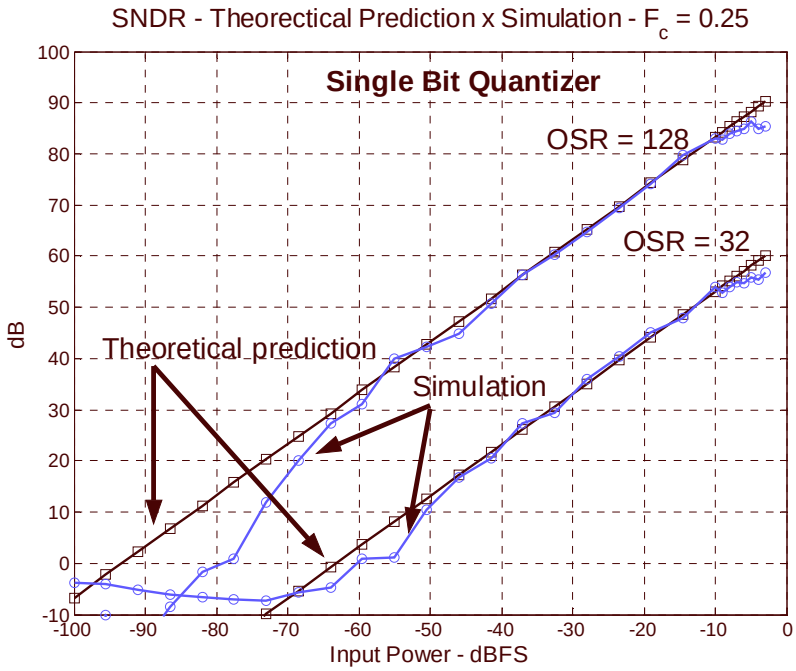


Figure 4.14: The **SNR** simulation and prediction for a BPΔΣMod of second order with a single bit quantizer.

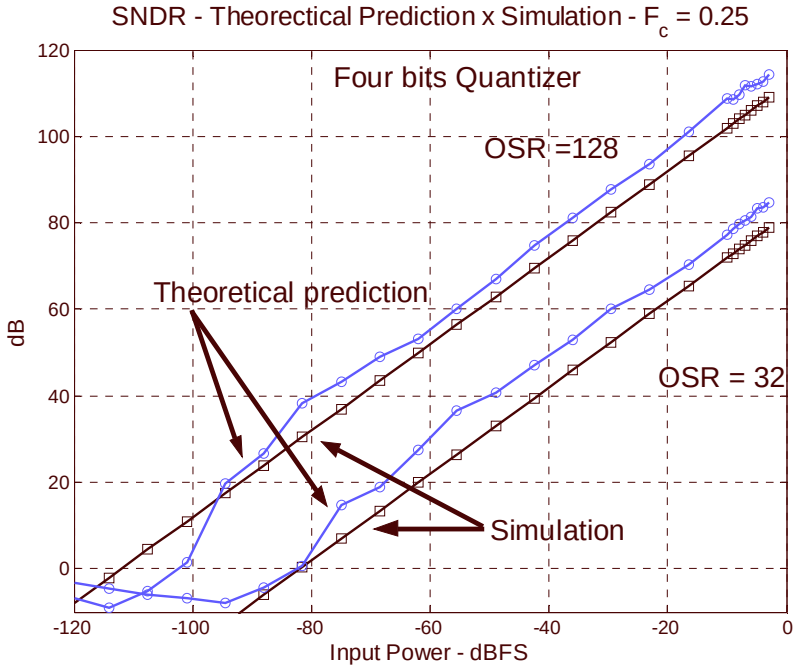


Figure 4.15: The **SNR** simulation and prediction for a **BPΔΣMod** of second order with a four bits quantizer.

4.2.4 Processing frequency f_p selection tradeoffs

In the last two sections, we presented the performance analysis and topology of the **FAC BPΔΣMod** and, according to equation 4.30, the ΔΣ **SNR** is independent of the

processing frequency selection. So, the pass-band of the ΔΣMod could be placed anywhere from **DC** to $\mathbf{f_s/2}$. Indeed, this is not true in real implementations. The resonator model used to derive the performance estimation equations has an infinite gain at Ω_P that is not achievable in real world circuitry mainly due to the finite gain product bandwidth (**GBW**) operational amplifier in active **RC** implementations or the transconductor output conductance in $\mathbf{g_m}$**-C** implementations [ORT2003, COR2003]. Additionally, it is interesting to recall that a 2nd order resonator with finite *merit factor* – **Q** has a -3dB bandwidth given by expression 4.32 and the **SBW** (equation 4.12) is function of both the **OSR** and sampling frequency ($\mathbf{f_s}$); therefore, for a constant **SBW** and moving down Ω_P this implies that the Q must go down in the same rate, making the value of 4.26 increase.

$$BW = \frac{\Omega_P}{Q} \tag{4.32}$$

On the other hand, from the application point of view, we can analyze the selection of the processing frequency from its implications at the input mixing process and at the **ΔΣMod**. For a given **SBW** and $\mathbf{f_P}$, the ratio selection $\mathbf{f_P/f_S}$ is a tradeoff between the sampling frequency (whole system speed), anti-aliasing filter requirements and **OSR**. Moving up $\mathbf{f_P}$ towards $\mathbf{f_S/2}$, the transition band for the required anti-aliasing is smaller, so in this sense moving down $\mathbf{f_P}$ would relax the anti-aliasing filter constraints. But, on the other hand, lowering the processing frequency makes more difficult the image rejection in the high input frequency range due to the mixing process and also, from equation 4.11, it increases the frequency dynamic range – **FDR** of the local oscillator, impacting on its complexity [VAZ2003].

To accommodate these tradeoffs among anti-aliasing and image-rejection filtering, the processing frequency should be placed at an intermediate frequency in the Nyquist band. An optimum frequency for it is $\mathbf{f_S/4}$. This choice offers a good balance in those filter constraints and also brings some additional advantages like a simpler design loop filter (analog) realization from **LP** to **BP** transformation and digital mixing processing to base-band is very simplified [VAZ2003].

4.3 Conclusion

This chapter presented the theoretical basis for predicting the **FAC** performance. The proposed modeling approach has divided the **FAC** noise and distortion calculation into two sources: the mixer and the ΔΣ modulator. The goal of this segmentation was to provide a way to account for the influence of each block in the **FAC** global performance estimation thus allowing more design space exploration at the topological level.

The mixer main noise source was considered as that generated by the digital control signals in form of time jitter as shown in equation 4.8. Actually, there are some additional noise and distortion sources that are present in the real mixer circuitry that must be accounted for at the mixer design time like thermal noise, port-to-port isolation, **IMD3** and **IP3**, for example; these parameters were already addressed as the metrics for evaluation of the mixer performance.

For the ΔΣ modulator block, the quantization noise was chosen as the main parameter to establish the relationship between the desired performance level and the **OSR**, quantizer number of bits and the **NTF** order. Again, the quantization noise is not the only performance degradation cause, but it reflects the intrinsic behavior of the ΔΣ

modulator. The designer has to deal also with the leakage in the integrators, the impairments in time constants, clock time jitter, the thermal and coupling noise in the active and passive devices. Most of these parameters are strongly correlated to the specific implementation topology that must be treated appropriately at design time [VAZ2003].

Therefore, the overall performance estimation of the proposed **FAC** is given by the worst **SNR** at the bit stream level, taking into account the contribution of the injected noise and distortion by the mixer and the **CTBPΣΔ** modulator. Since the **CTBPΣΔ** modulator has its intrinsic performance independent of the location of the input signal in the spectra, the frequency **SNR** degradation comes from the mixer block and local oscillator timing block that gives the frequency coverage limitation, as stated by equation 4.8. A general expression for the **FAC** performance can be drawn using the noise and the induced distortion contribution of each individual block. Equation 4.33 expresses the total **SNR** of the proposed **FAC**, where $\mathbf{P_i}$, $\mathbf{P_{QN}}$ and $\mathbf{P_M}$ are the input signal, the quantization noise and the mixer noise power, respectively.

$$SNR_{FAC} = \frac{P_i}{P_{QN} + P_M} \tag{4.33}$$

Equation 4.33 synthesizes the achievable **FAC SNR** and it will be used in the next chapter to perform a design space exploration for a **FAC** design parameter determination to achieve a desired performance goal.

5 THE FIXED ANALOG CELL DESIGN

The previous chapter addressed the *fixed analog cell* – FAC model and performance prediction that resulted in equation 4.33. From this equation, two noise contribution sources are identified, one from the mixer and the other one from the **BPΣΔ** modulator. Figure 5.1 illustrates the **FAC SNR** behavior as the mixer switching frequency increases for different combinations of ΔΣ modulator orders, oversampling ratios and time jitters. The RMS values of the time jitter are normalized to the highest frequency $\mathbf{F_{max}}$ to be processed by the interface corresponding to 1% and 0.1% of the maximum mixer operating frequency $\mathbf{F_{max}}$ period. The frequency analysis span is also normalized, but now to the processing frequency $\mathbf{f_P}$ leading to a maximum mixer frequency of 100 times $\mathbf{f_P}$. There are two distinct regions in the **FAC SNR** behavior over the frequency, one flat and another with an **f** falling rate. In the first one, in the low frequency range, it is clearly predominant the **ΔΣMod** induced quantization noise and distortion, so the **OSR** and the number of resonators **N** impact the overall performance. As input frequency increases the noise power generated by the mixer source clock becomes the limiting performance factor. Therefore, independently of the system order or the oversampling ratio, the interface performance remains limited.

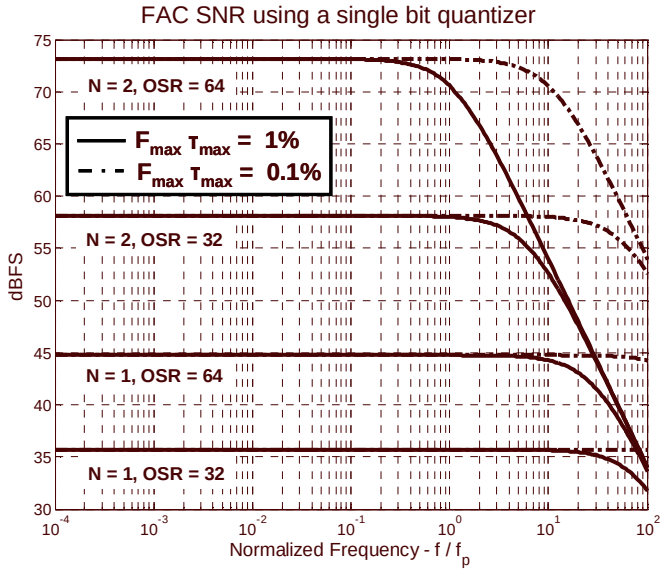


Figure 5.1: The relative mixer time jitter, the resonator order and the **OSR** frequency impact over the **FAC** signal-to-noise-ratio as a function of input signal frequency.

Figure 5.1 also clearly illustrates that for applications in the flat region the limiting, performance comes from the noise and distortion in the **ΔΣMod**, as we stated above. So, some comments must be done regarding the **ΔΣMod** design phase and the application mapping at the configuration phase.

First of all, before we go deep inside the **FAC** design example, a few points are necessary to discuss concerning the tradeoffs that emerge from the **FAC** performance analysis. Taking the highest target bandwidth specification, corresponding to the signal bandwidth **SBW**, and the required **SNDR**, these constraints set the bottom of the **ΔΣMod** performance. So, during the **FAC** design time, this constrains are used to balance the modulator complexity and the sampling frequency. It must be refreshed that the complexity of ΔΣ modulators come mainly from the selected order **N** (number of cascaded resonators) and the quantizer number of bits **b** (resolution). By increasing the number of resonators a more aggressive noise shaping is obtained. Alternatively, increasing the quantizer number of bits a total quantization noise power reduction is achieved in the loop filter. Both approaches lead to a higher final **SNDR** as shown by equation 4.26. The best design should be the one with the lowest sampling frequency, the highest **OSR**, the lowest number of resonators and quantizer bits that satisfies both the required signal bandwidth **SBW** and **SNDR**. The sampling frequency impacts on the power consumption not only of the analog blocks but also of the digital control and processing blocks, so using the lowest sampling frequency will reduce the power consumption. Augmenting the **OSR** results in **SNDR** improvement, but the sampling frequency is related to the signal bandwidth by the **OSR** (equation 4.27), then to maintain a constant **SBW** the entire modulator clocking scheme boosting up is necessary what can bring timing related issues like time jitter that degrades the **SNDR** [CHE99] and, additionally, it increases the power consumption. Finally, the lowest number of resonators and quantizer bits is desired because they are related to modulator complexity, enhancing the modulator complexity results in larger area consumption.

5.1 The FAC Specification and Design

To illustrate the above reasoning, let us suppose a general purpose **SOC** analog interface that will be used to process signals from **DC** up to 20 MHz, with bandwidth of at most 60 kHz, with achievable output **SNR** of 50dB (**ENoB** = 8 bits). The proposed **FAC** could be used as this interface and we will show how it can be designed.

Looking at figure 5.1, the **FAC SNR** over frequency resembles the low pass filter response. Then, we could also define a mixer cut off frequency where the noise power of the two sources have the same magnitude, this point corresponds to the -3dB point in the performance curves in figure 5.1. The mixer cut off must be equal to the highest input center frequency band. This way, the expressions 4.8 and 4.33 are applied to determine the maximum tolerable mixer clock source **RMS** time jitter, that is approximately 16 ps. Setting **k** equal to 0.25 ($\mathbf{f_p = f_s/4}$) to force the lowest sampling frequency and, employing equations 4.26 and 4.28, it results that a 4^{th}-order **BPΣΔ** modulator using a 1 bit quantizer operating with **OSR** equal to 32 could deliver the suitable **SNR**, hence the $\mathbf{f_P}$ and $\mathbf{f_S}$ will be 1 MHz and 4 MHz, respectively. The choice of single bit quantizer regards the intrinsic linearity of such **DAC** and **ADC**. According to figure 4.9, the discrete prototype for the proposed solution is shown in figure 5.2. Indeed equation 4.33 predicts an upper performance bound, because some other secondary effects that emerge in real implementations were not taken into account at this time, but they are well-known and also extensively treated in literature related to ΔΣ

modulators [CHE99, ORT2003]. This way, it is mandatory to leave some performance headroom to accommodate the leakage in the integrators, the impairments in time constants, control clock time jitter, the thermal and coupling noise in the active and passive devices [VAZ2003], for example.

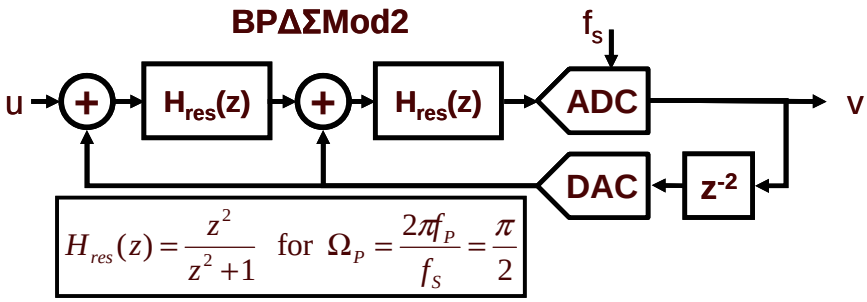


Figure 5.2: The discrete prototype architecture for a 4th order band pass ΔΣ modulator with center frequency at **π/2**.

5.1.1 The continuous time band-pass ΔΣ modulator architecture

From the discrete time loop filter prototype, the continuous time equivalent loop filter must be determined using the procedure defined by equation 4.21. From the analysis of equation 4.21, it is mandatory to know the exact shape of the feedback **DAC** pulse that is going to be used to close the loop and excite the resonators. Actually, there are two basic **DAC** rectangular pulse shapes, one called *non return to zero* – **NRZ** and another called *return to zero* – **RZ**, as figure 5.3 shows. The **NRZ** pulse shape remains at the same reference voltage/current level from the beginning till the end of sampling period $\mathbf{T_s}$. The other case, the **RZ** pulse shape, it stays at the reference voltage/current level from the beginning till time $\mathbf{T_D}$ when it changes from the reference current/voltage level to zero.

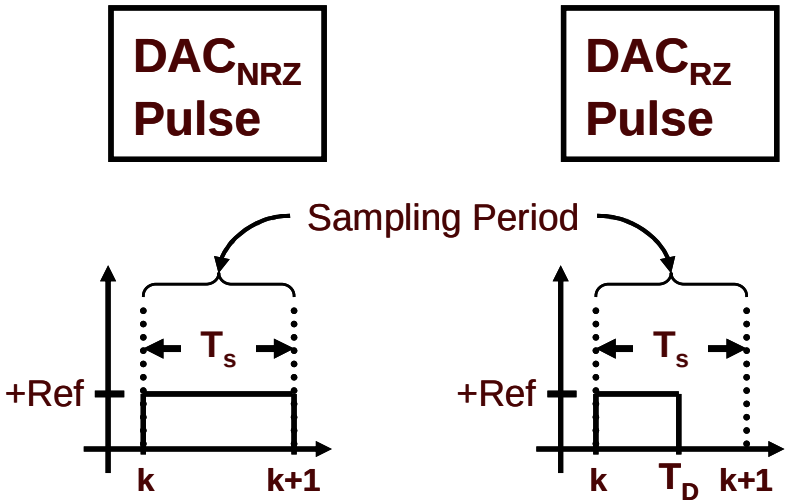


Figure 5.3: The **NRZ** and **RZ DAC** pulse shapes.

The **NRZ** approach is easier to design since it is indeed the pulse invariant inverse **z** transform that is implemented in tools like Matlab®. Equation 5.1 shows the continuous time loop filter using this approach; it is normalized to $\mathbf{f_S}$. Four feedback paths are necessary to create the zeros of this loop filter using the classical state variable

topology, each one of the feedback paths goes to one state variable or integrator, so in each integrator we need a summing node. From the implementation point of view, a **NRZ DAC** pulse suffers of memory effects due to finite bandwidth and rise and fall times of the **DAC** drivers, that is more severe for signals near the full scale when long sequences of zeros or ones appears [VAZ2003]. On the other hand, the **RZ DAC** pulse shape a return to zero state always occurs, so eliminating the memory effects.

$$NTF_{NRZ}(s) = \frac{0.25 \cdot s^3 - 0.1685 \cdot s^2 + 0.6169 \cdot s + 1.522}{\left(s^2 + \pi^2/4\right)^2} \tag{5.1}$$

The selected continuous time topology is depicted in figure 5.4. We adopt the **RZ** scheme as shown. Actually, two different **RZ DAC** pulse shapes with duration of half the sampling period were used, one beginning at the sampling instant ($\mathbf{DAC_{RZ}}$) and the other half sampling period delayed ($\mathbf{DAC_{HRZ}}$). Besides the elimination of the memory effects, this approach creates the possibility to use only two summing nodes as figure 5.3 presents [CHE99]. One of the costs of this approach is that the discrete time to continuous time transformation (equation 4.21) is much trickier than usually is, because both pulses last less than a sampling period, the conventional **z** transform approach cannot be used. Instead the *modified* ***z*** *transform* must be applied, because it incorporates a delay with magnitude less than a sampling period in its definition. Using this approach the feedback gains, according to figure 5.4, $\mathbf{H1_{HRZ}}$, $\mathbf{H2_{HRZ}}$, $\mathbf{R1_{RZ}}$ and $\mathbf{R2_{RZ}}$ are 2.987, 1.087, -0.6339 and -0.4502 respectively. Those gains were calculated using Maple® symbolic tool. The details of this development are described in Appendix I.

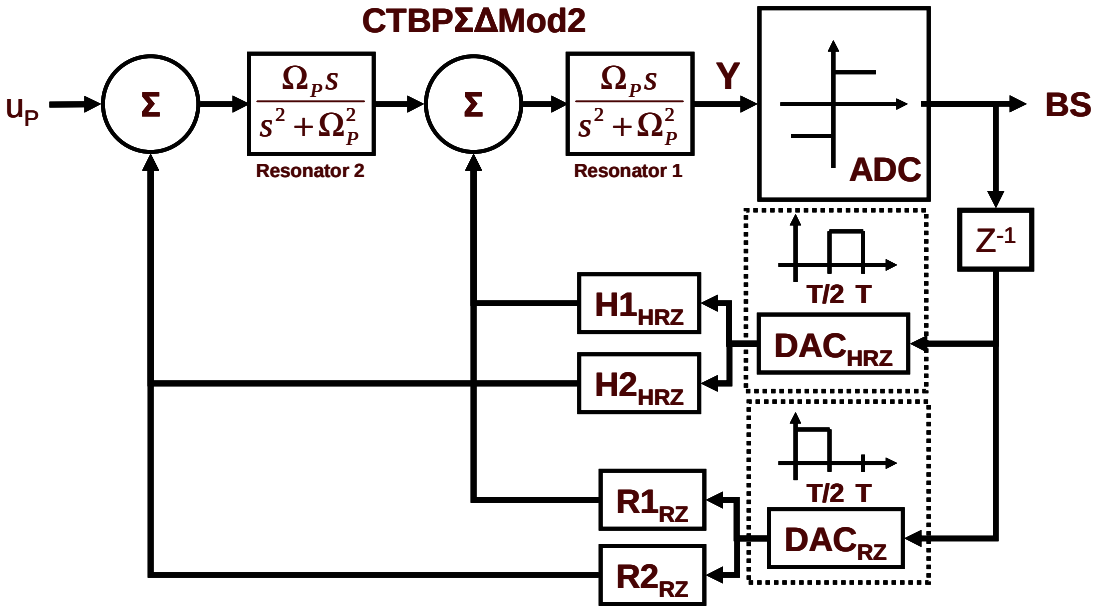


Figure 5.4: The continuous time band-pass ΔΣ modulator architecture.

As shown in figure 5.4, the one sample time delay between the **ADC** and the feedback **DAC** was maintained to guarantee a sufficient conversion time to the **ADC**.

The continuous time ΔΣ modulator architecture definition makes possible the development of a simulation model to draw comparisons between the **CTBPΔΣMod** against the theoretical **SNR** prediction and the **CTBPΔΣMod** against the digital seed

prototype, as start validation point for the modeling and design approach employed. Therefore, the **CT** ΔΣ modulator described above was modeled using the Simulink® and Matlab®; the continuous time resonators were modeled using state variable filter architecture with ideal integrators, as shown in figure 5.5. Actually, the model construction has at least two reasons, one regarding the validation step and the other concerning the prototype design phase and component selection.

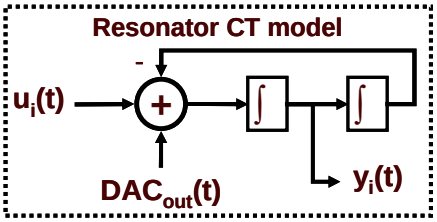


Figure 5.5: The resonator **CT** simulation model.

The built **CT** model was used to run several simulations using a cosine input signal with power ranging from input full scale down to -100 dBFS; for each **OSR** the signal frequency was set to be apart 2/3 of **SBW** from processing frequency $\mathbf{f_P}$. The **SNDR** comparison was done using the theoretical prediction model, developed in the last chapter, and the **DT** simulation; in these simulations only the quantization noise and distortion induced by ΔΣ modulator were considered. The results are shown in figure 5.6 for an **OSR** equal to 32, 64, 128 and 256. Figure 5.6a demonstrates a good agreement between the theoretical prediction model and **CT** simulation results; a greater deviation occurs above -6 dBFS inputs where the quantizer starts overloading and the linear model fails, as mentioned earlier. Moreover, the **DT** to **CT** mapping depicted earlier seems to be done correctly, since the **SNDR** from the **CT** simulation and **DT** closely match. Therefore, the **CT** ΔΣ modulator proposed architecture is validated and able to go to next design phase comprising the definition of all its components.

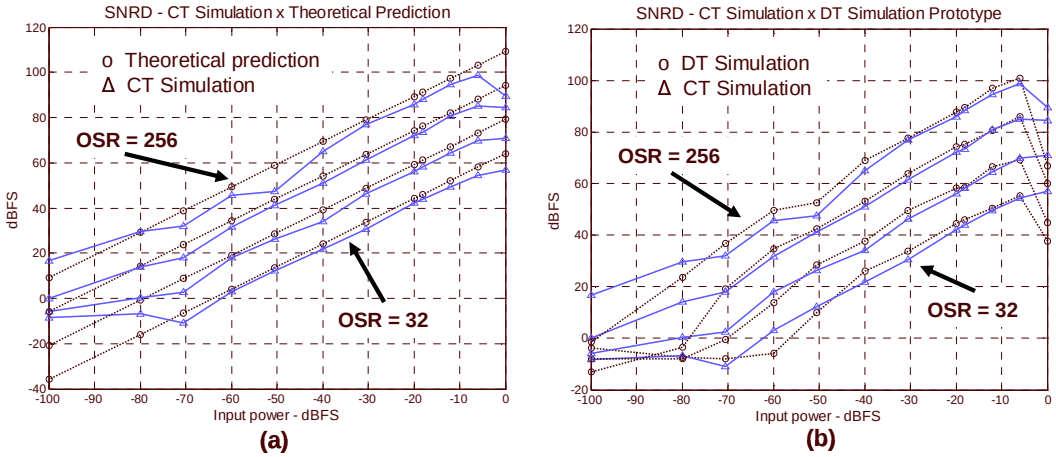


Figure 5.6: The SNDR comparison between for the CT simulation model against (a) the theoretical prediction model and (b) the DT prototype simulation for an **OSR** equal to 32, 64, 128 and 256.

In the next section, some of the design steps that are important to be done before the implementing the **FAC** prototype using discrete or integrated real devices are shown.

5.1.2 The FAC prototype design

Our objective is to design a **FAC** that could fulfill the application constraints as described earlier, using off-the-shelf components. So, it is necessary to select the circuit topology implementation after the high level architecture validation.

For implementing the continuous time filters there are basically two approaches, one technique using transconductors and capacitors that is the well known $\mathbf{g_m}$**-C** approach, and another one known as active-**RC** approach, that employs operational amplifiers. The $\mathbf{g_m}$**-C** technique uses open loop transconductors for implementing the state variable of the **CT** filter that are more susceptible to harmonic distortion, but they can achieve higher frequencies. On the other hand, the active-**RC** approach uses operational amplifiers with local feedback that minimizes harmonic distortion by the loop gain, but it is required that the gain-bandwidth product of the operational amplifier must be much larger than the required filter frequency response. The active **RC** approach was chosen to implement the **FAC CTBPΔΣMod** prototype because it leads to a low distortion implementation, and also because it is much easier to find operational amplifiers than transconductors for a discrete components implementation [PAT2004].

The **CTBPΔΣMod** design faces additional constraints regarding the frequency response and dynamic range of the active device used, so those parameters must be taken in account. Initially, the simulation model used for checking the **DT** to **CT** transformation was also employed to determine each integrator necessary dynamic range. The integrators **DR** information was used to compute the impedance scaling, so all four integrators of the resonators should have almost the same output **DR**. Another aspect related to scaling are the feedback **DAC** gains, actually those gains in combination with the reference voltage value define the absolute maximum range in the integrators, so they were also scaled to accommodate the actual operational amplifier maximum input and output swing.

The same simulation model was improved in a way to also account for the operational amplifiers frequency response. Figure 5.7 consolidates this procedure showing the ideal and the expected **CTBPΔΣMod PSD** considering integrators impedance and reference voltage scaling and finite product gain bandwidth operational amplifier. Analyzing the simulation result of figure 5.7a a performance loss is observed near $\mathbf{f_P}$, where the **PSD** becomes flat. This flatness will negatively impact the **SNR** performance when increasing the **OSR** by reducing the 15 dB/Octave rate as predicted for a 2^{nd} order **CTBPΔΣMod,** as depicted by figure 5.7b. This will occur for a signal which bandwidth falls in the flat region. The flatness in the **NTF** owes to a reduction in attenuation by the noise shaping notch filter. The scaling does not affect the overall behavior of the **CTBPΔΣMod**, actually the performance loss is due to the active devices finite gain that it resembles the effect also observed in low pass ΔΣ modulators implementations, either **DT** or **CT** [ORT2003]. The simulation results shown in figure 5.7 indeed were conducted using the actual **GBW** data of the selected operational amplifier which is 240 MHz what gives a gain around 200 in 1MHz.

The sampling frequency $\mathbf{f_S}$ was actually set to 4.096 MHz instead of 4 MHz, because the FPGA (Altera® ACEX1K EP1K100QC208 device) board main clock used for implementing the control logic is 32.768 MHz, so it is straightforward to make a frequency divider with a factor of 8; this leads to a processing frequency $\mathbf{f_P}$ equal to 1.024 MHz.

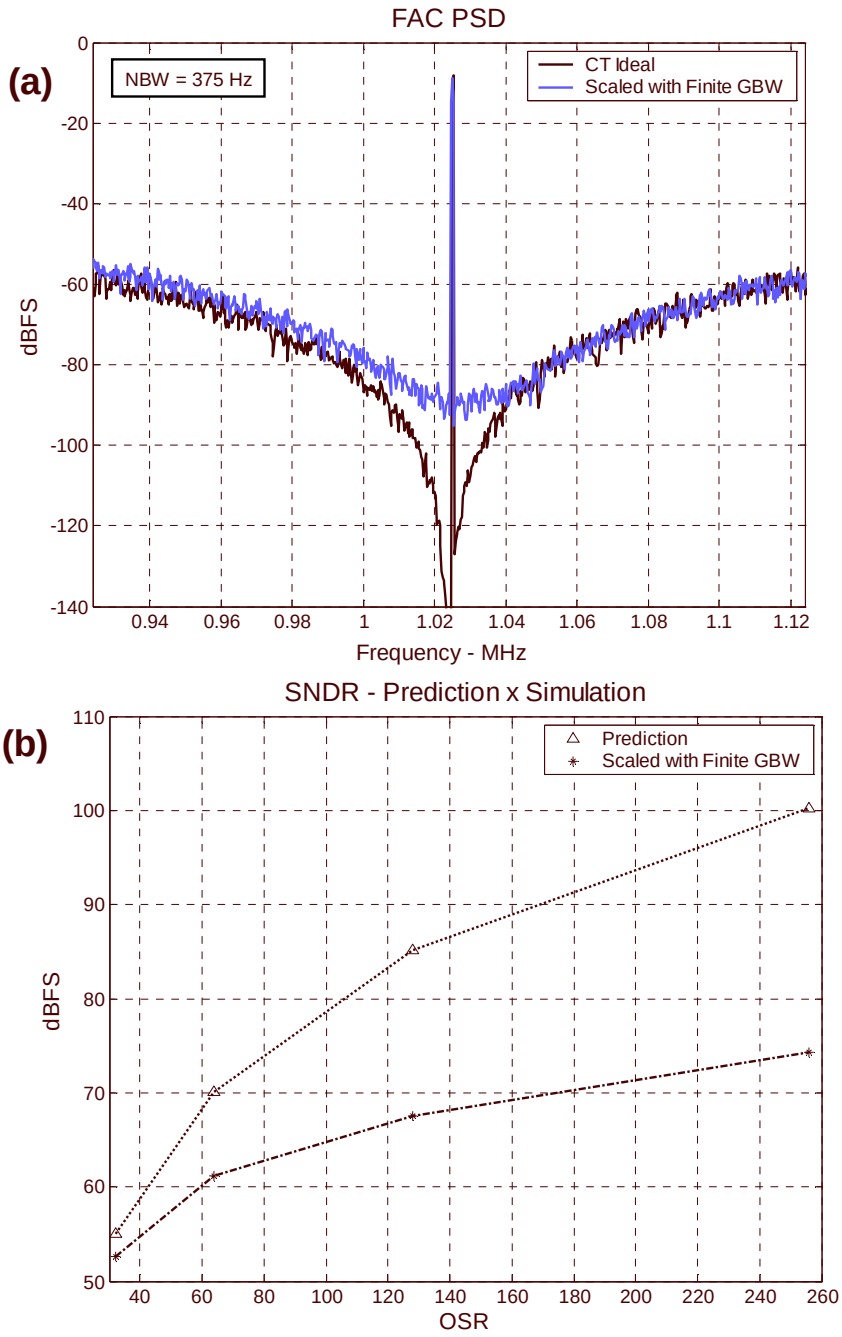


Figure 5.7: The ideal **CTBPΔΣMod PSD** and considering finite product gain bandwidth operational amplifiers: (a) **PSD** and (b) **SNDR** versus **OSR**.

The mixer block was implemented using the analog CMOS switches SN74LVC2G66 [TEX2003] that uses transmission gate topologies. The main characteristics of these switches are the turn on and turn off times around 0.5 ns, on resistance of 6.5 Ω and 'on' state bandwidth above 300MHz.

The next section will discuss the prototype characterization and additional details about the prototype design are presented in Appendix I.

5.2 The Fixed Analog Cell Evaluation Results

A fixed analog cell discrete prototype was built to provide support for the proposed methodology and, also, to demonstrate the application mapping potentiality of this interface. This prototype comprises a **FAC** and a digital infrastructure based on a **FPGA**. The fixed analog block was built as stated in the last section. The **FAC** was combined with a FPGA board with an ACEX1K EP1K100QC208 device. This FPGA board has an internal clock source of 32.768 MHz that limits the maximum mixer control frequency to half of this clock or 16.384 MHz, so the input frequency span is from DC to 17.408 MHz as the processing frequency is 1.024 MHz. Although discrete, the overall principle is certainly valid for any VLSI implementation, with the ensuing benefits. Appendix I shows additional details regarding this implementation, some of this details are model construction for prefabrication validation, impact of real world components imperfections on the **FAC** overall performance and prototype debugging.

After building the prototype, a series of testing procedure were conducted to evaluate the intrinsic performance of the system to verify its compliance with the application whish list and the performance prediction. The evaluation procedure was divided in three steps with the following objectives:

Step 1 - the ΔΣ modulator noise shaping characterization and system distortion;

Step 2 - the frequency response and the achieved performance;

Step 3 - the **DC** linearity evaluation.

Figure 5.8 shows the testing setup used to carry out the evaluation process. Depending on the characterization step, the appropriate signal sources or analyzer are connected to the prototype. The Agilent mixed-signal (**MS**) is used as primary data collector, since it can acquire analog and up to 16-bit digital signals synchronized with the analog section. Also, the **MS** oscilloscope has a time jitter analysis application module that enables it to make time jitter measurements on the two analog channels. The computer purposes are the instrumentation control and data acquisition, as well as data analysis post-processing under Matlab®.

In the following, the results obtained in each one of the above testing steps are discussed.

5.2.1 The ΔΣ Modulator noise shaping characterization and distortion

The objective of this evaluation test is to measure the matching between design and the prototype noise shaping and distortion properties of the whole interface using the **FAC**. Actually, this procedure was partitioned into three tests: one regarding the **NTF** characterization, and two related to intermodulation and distortion effects.

To characterize the ΔΣ modulator noise shaping, a 9.217 MHz single tone input frequency that corresponds to a signal 1 kHz apart from processing frequency $\mathbf{f_P}$ when the local oscillator frequency (mixer control) is set 8.192 MHz. The resulting digital bit-stream was captured by the oscilloscope, the instrumentation setup is depicted in figure 5.9. The bit-stream record length is about **1 s** long (4.1 M points) that corresponds to the maximum oscilloscope memory depth.

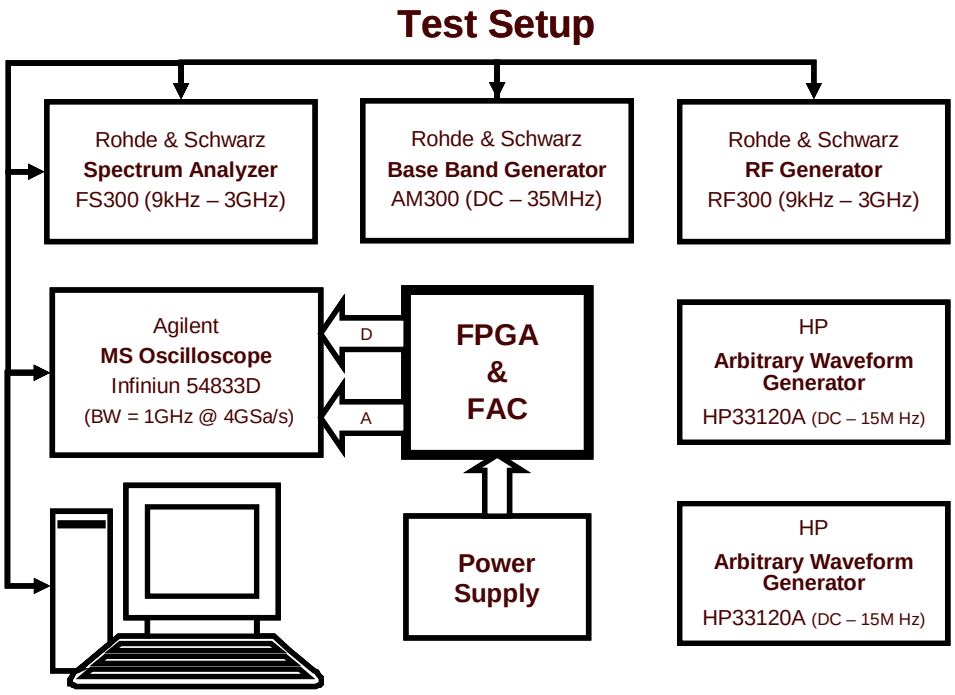


Figure 5.8: The instrumentation setup used to evaluate the prototype performance.

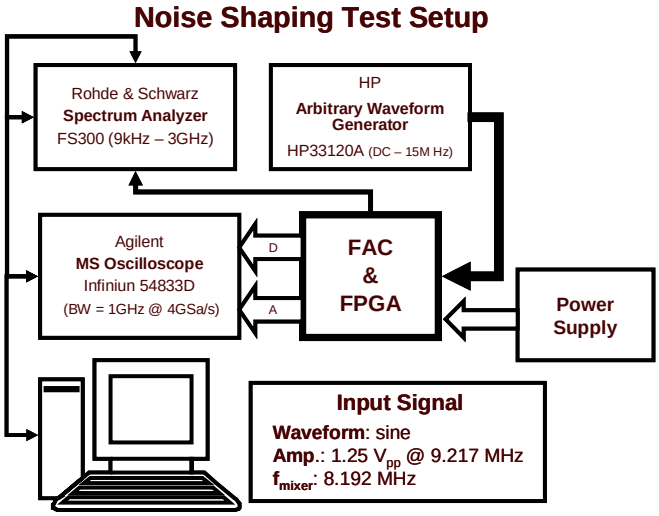


Figure 5.9: The **NTF** characterization instrumentation setup.

The acquired bit-stream was processed to determine its mean square average *power spectrum density* – **PSD** using *Hanning* windowing with 2^{17} points, what gives a frequency resolution of 31.25 Hz and an equivalent *noise bandwidth* – **NBW** of 46.9 Hz. Then, the resulting **PSD** was compared with the ideal **CT** model and the one taking into account real devices and system characteristics. The result is shown in figure 5.10. The ideal **CT** case corresponds to simulation **A**, the simulation **B** considers the operational amplifier finite **GBW** and time jitter in the mixer control signals and the last one corresponds to the measured **NTF** from the prototype. All **PSD** was calculated using the same **NBW**, so comparisons can be done directly [LAT98]. The operational amplifier finite **GBW** provokes the flatness in the middle of the ΔΣ modulator pass-band, however it is not the only cause for this effect, a ΔΣMod clocking scheme time jitter can also affect in the same way the noise shaping characteristic, as Cherry and Tao

reported in [CHE99] and [TAO99a] respectively. Equation 5.2 gives the total noise power over the signal bandwidth due to the time jitter in the **RZ** pulse shape **DAC** feedback lines where $\mathbf{T_{DAC}}$ is the **DAC** pulse width time, $\mathbf{\Delta}$ is **DAC** pulse step and $\boldsymbol{\tau_{RMS}}$ is the **RMS** time jitter value [TAO99a].

$$P_\tau \approx \frac{2}{OSR}\cdot\left(\frac{1}{f_S\cdot \mathrm{T}_{DAC}}\right)^2 (\tau_{RMS}\cdot f_S)^2 \frac{\Delta^2(\pi\cdot f_P\cdot \mathrm{T}_{DAC})^2}{\sin^2(\pi\cdot f_P\cdot \mathrm{T}_{DAC})} \tag{5.2}$$

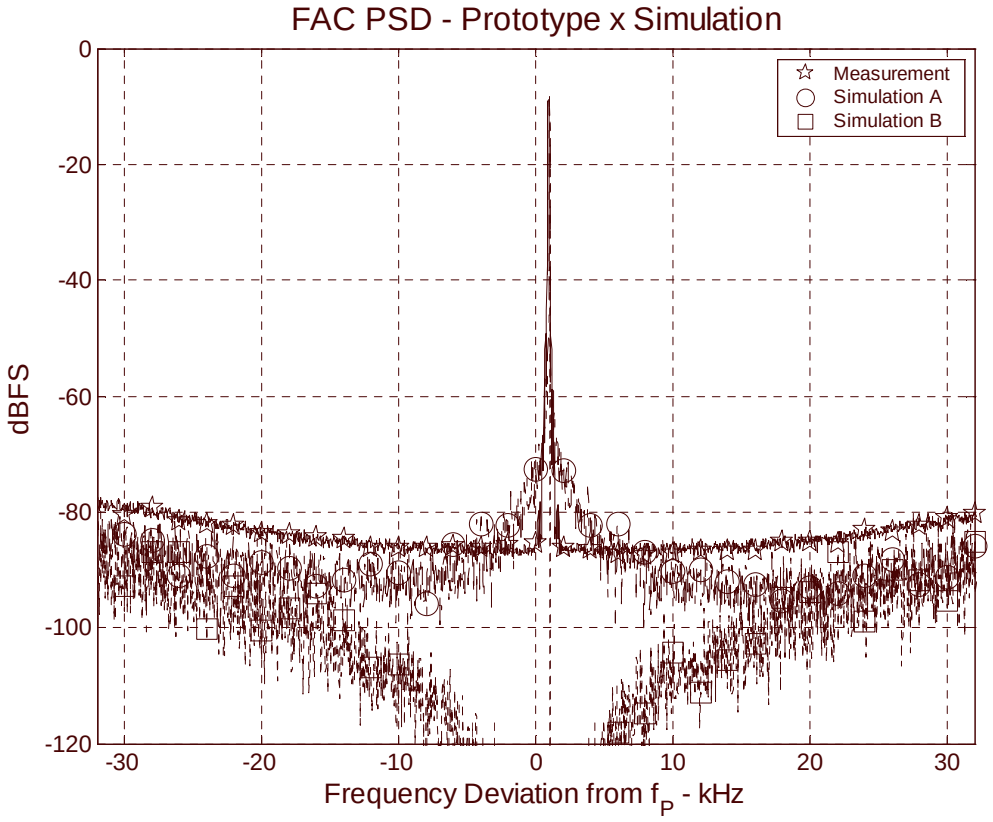


Figure 5.10: The **FAC PSD** comparison between a full scale single tone input signal at 9.217 MHz (Measurement) and an ideal **CT** model simulation (Simulation B) and a **CT** model taking in account finite GBW of operational amplifier and mixer time jitter (Simulation A).

Equation 5.2 enables one to establish the maximum achievable performance when the dominant noise source is the time jitter. Therefore, considering a prototype full scale single tone input, the maximum achievable ΔΣ modulator **SNR** is stated by equation 5.3 when jittered control signals are present and overriding the quantization noise.

$$SNR_{Max} = 10\cdot\log\left(\frac{P_S}{P_\tau}\right) \approx 10\cdot\log\left[OSR\cdot\left(\frac{\sin\left(\frac{\pi}{8}\right)}{\pi\cdot\tau_{RMS}\cdot f_S}\right)^2\right] \tag{5.3}$$

$$\text{where}\left\{P_S=\frac{A^2}{2},\ \mathrm{A}=\frac{\Delta}{2},\ f_P=\frac{f_S}{4}\ \text{and}\ \mathrm{T}_{\mathrm{DAC}}=\frac{1}{2f_S}\right.$$

Using the Infiinium ® oscilloscope, the time jitter **RMS** value was characterized for each relevant control signal generated by the prototype **FPGA**. Therefore, jitter induced

in **DAC** pulses were found to be 240 ps_{RMS}, which establishes a **SNR** limit of 65.9 dBFS for an **OSR** of 256.

The following step was the distortion characterization that was done by injecting an input signal with the following characteristics:

1. an amplitude modulated signal (AM) with modulation index **m** equal to 100%
2. the carrier signal frequency $\mathbf{f_c}$ equal to 6.490 MHz;
3. a sine waveform modulating signal with frequency $\mathbf{f_m}$ set to 10 kHz;
4. the mixer frequency was programmed to 5.4613MHz.

Equation 5.4 shows the mathematical description of an **AM** signal such the one described above.

$$in(t) = A \cdot [1 + m \cdot \sin(2\pi \cdot f_m \cdot t)] \cdot \sin(2\pi \cdot f_c \cdot t)$$
$$\text{where } \{A = A_{\max},\ m = 100\%,\ f_c = 6.490MHz \text{ and } f_m = 5kHz \tag{5.4}$$

The bit-stream was acquired and processed to find the output **PSD** exactly in the same way used to evaluate the **NTF**, the result is shown in figure 5.11a, where the frequency scale is set to a frequency deviation from $\mathbf{f_P}$. The injected signal generates theoretically 3 bins in spectrum: one in the carrier frequency and two carrier side bins $\mathbf{f_m}$ apart from the carrier frequency. The measurement of the input spectrum in figure 5.11b shows at least two additional bins apart 2 times $\mathbf{f_m}$ that actually correspond to the generator third harmonic intermodulation characteristic **IMD3**. So, with the instrumentation available at the measurements period it is possible to state, that the **FAC IMD3** is higher than 60 dB, and its corresponding *third harmonic input intercept point* – **IIP3** is greater than 15 dBFS, since it reproduces exactly the input excitation inside the signal bandwidth hitting the instrumentation limit.

In the next section the **DC** behavior characterization is going to be addressed.

5.2.2 The DC behavior evaluation

One of FAC potential applications is in the control and instrumentation field, thus for such purposes the interface linearity is one of the most important factors. Actually, most of application in this field deal with small bandwidth signal sources like in temperature and pressure measurements, for example. This way, one approach to characterize the low frequency linearity of the **FAC** is to excite it with a very low frequency ramp and analyze its output bit-stream.

The test strategy employed was to excite de **FAC** with a near **DC** triangle waveform with frequency and amplitude equal to 2 Hz and 1.5 V_{pp}, this implied the mixer frequency set to 1.024 MHz. The input (analog signal) and the generated bit-stream (digital) were acquired by the oscilloscope. The instrumentation setup employed in this test procedure was quite the one shown in figure 5.9 except that the spectrum analyzer was not used.

Figure 5.12 illustrates the mathematical processing flow employed to analyze the **FAC** linearity using the collected data [SHE86]. The bit-stream and the analog input triangular signal were acquired using the **MS** oscilloscope and sent to a computer for processing.

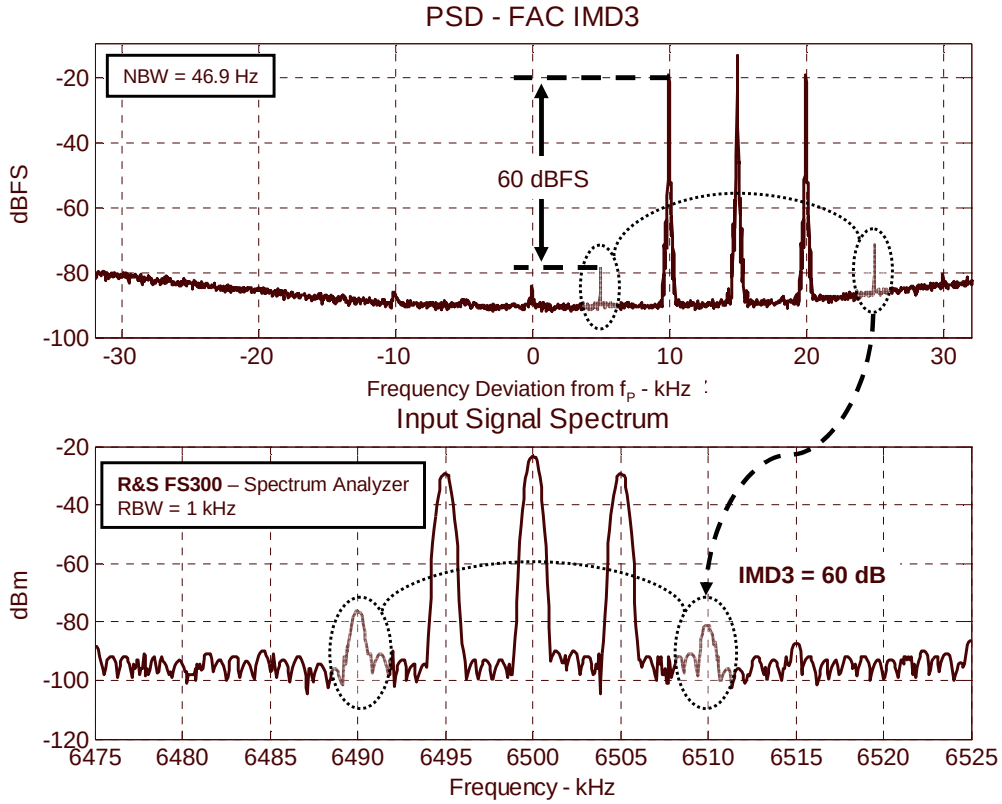


Figure 5.11: The comparison between (a) an acquired bit-stream **PSD** with an amplitude modulated **FAC** input signal and the corresponding (b) input signal spectrum measurement.

Using Matlab® for data analysis, the acquired signals were first submitted to a down sampling repeatedly until the same 1 kHz output sampling frequency $\mathbf{f_{Sout}}$ resulted for both the **FAC** and the oscilloscope signals. As the input sampling rate of each device was different the acquired signals were submitted to different down sampling rates **R**, but the decimation filter constraints were specified for both signals accounting for the effective output sampling frequency of each decimator stage; the decimation rates **R** employed in each signal path are shown in the flow processing in figure 5.12.

After the decimation, both signals had the offset cancelled and they were also normalized to **1** unit peak, as shown in figure 5.13a that seems both signals have a close match. But, looking to figure 5.13b that magnifies both signals around zero crossing, the triangular wave acquired through the scope is bouncing around **FAC** signal. Therefore, following the normalization the segmentation in the fall and rise semi-periods was done and the segmented **FAC** data was used to make two linear regression sets using the mean square error minimization algorithm either for the rise and fall sections.

The first regression set was done taking as standard signal the processed input signal acquired by the oscilloscope. The error (figure 5.13c) and its power were determined using the fitting coefficients that make possible to calculate the equivalent number of bits (**ENoB**) using 4.13, resulting in 8.7 bits. Actually this **ENoB** matches the oscilloscope **ADC** ½ bit resolution that for the sampling rate of 2 MSa/s is 8 bits.

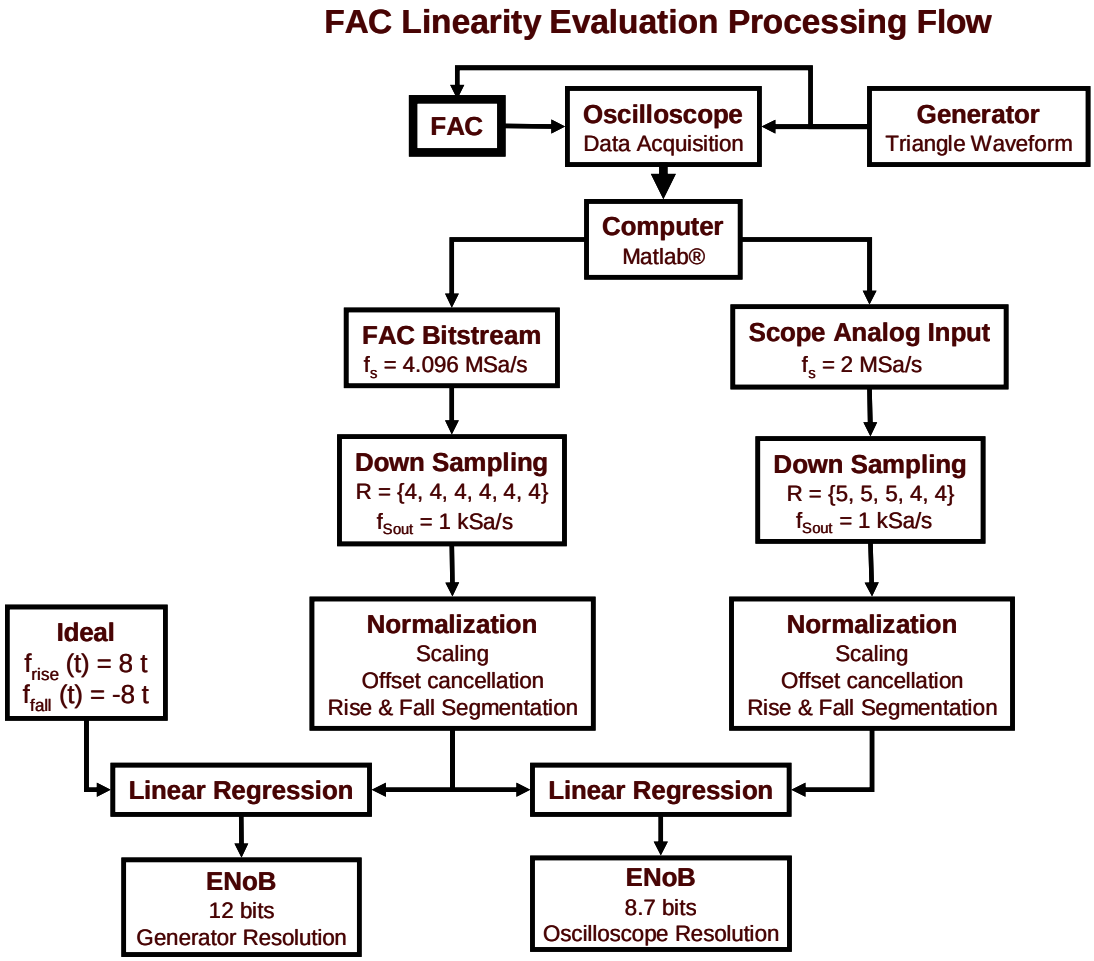


Figure 5.12: The processing flow employed to evaluate the **FAC** linearity using a 2 Hz triangle input waveform.

The second regression set uses an ideal straight line with rate equal to the input normalized rate of 8 units per second for the rise segment (-8 for the fall segment) as pattern, the error (figure 5.14) and its power were also determined and the **ENoB** was calculated resulting in 12 bits at this time.

Actually, the last result reaches the linearity limitation of the input signal source that is 12 bits according to its specification [AGI2004]. However, if equation 5.3 is used to calculate the maximum achievable **SNR** in the presence of a time jitter of 240 ps_{RMS}, as reported in the last section, for the employed **OSR** of 4096, it results that $\mathbf{SNR_{max}}$ is equal to 78 dBFS or 12.7 bits. From this result, it is possible to conclude that the analysis procedure achieved both the theoretical and the experimental limits.

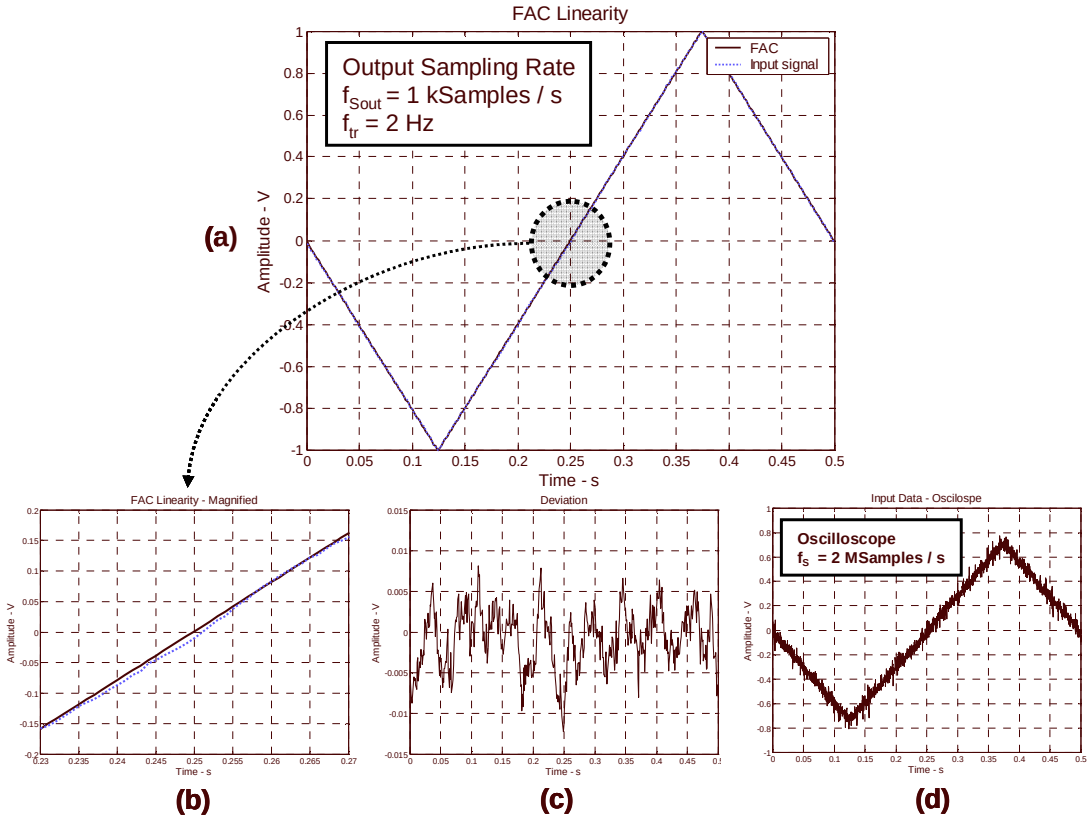


Figure 5.13: The time domain **FAC** linearity test signals, in (a) both input and FAC signals scaled, (b) input signals magnified around zero crossing, (c) the regression error using the oscilloscope as pattern and (d) the pre-processing input signal from the oscilloscope.

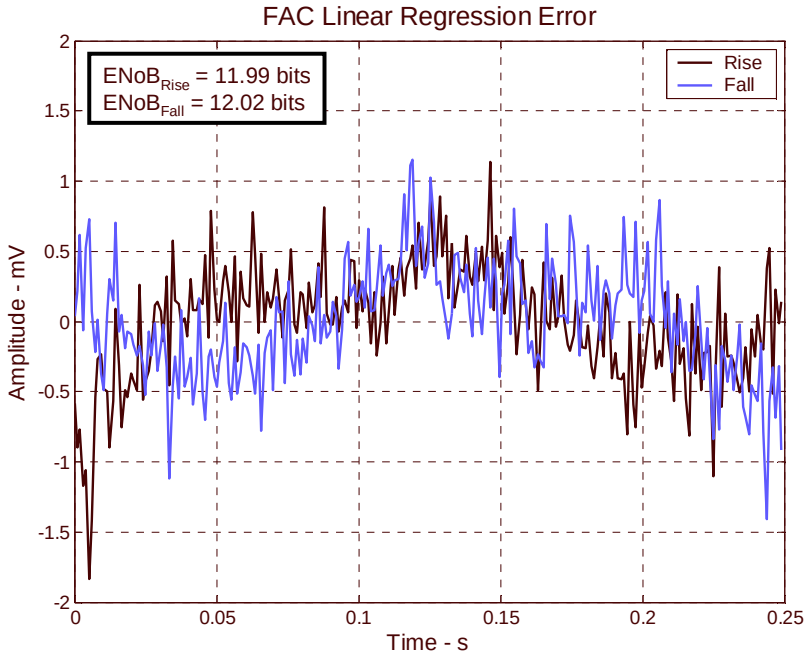


Figure 5.14: The FAC signal amplitude error when compared to an ideal triangular waveform with the same frequency and amplitude.

5.2.3 The frequency response evaluation

As this interface was designed to treat signals from **DC** to high frequency, it is important to characterize the **SNR FAC** behavior over frequency. This test was done by applying a full scale single tone at the interface input and the corresponding bit-stream was acquired for each testing frequency. The input frequencies were set such that they create an image tone apart about 2/3 **SBW** from $\mathbf{f_P}$, so this test signal would be near the signal bandwidth edge. It is interesting to reinforce that the signal bandwidth is a function of the **OSR** as equation 4.23 states.

For each input signal, the bit-stream was processed to determine its mean square average *power spectrum density* – **PSD** using *Hanning* windowing with 2^{17} points, using the **PSD** information, signal bandwidth and the tone position in the spectra, the **SNR** is computed. The results are shown in figure 5.15. Actually, figure 5.15 shows both the achieved **SNR** for each input frequency and **OSR**, and the theoretical prediction given by equations 4.33 with the additional noise source due to the time jitter present in ΔΣ modulator **DAC** control lines as expressed by equation 5.2. The time jitter measurements of the mixer and of the ΔΣ modulator were used in the prediction equations, time jitter measurement was 240 ps_{RMS}.

Figure 5.15 shows the main prototype measurements results since it synthesizes the whole **FAC SNR** performance from **DC** up to 17.4 MHz for the **OSR** equal to 32, 64, 128 and 256. The measurements show that the **FAC SNR** drops when the frequency increases, the explanation for this behavior comes from the presence of time jitter in the mixer control lines with **RMS** value practically constant from low to high frequency, so more noise induced by the mixer is present as the frequency rises up, like equation 4.8 states. Analyzing the results for the **OSR** from 64 till 256, the theoretical formulation predicts the **FAC** behavior over frequency quite well with an error close to -3 dB. But, in the case for an **OSR** equal to 32, the prediction error is higher reaching a deviation around -8 dB in the worst case. The explanation for this discrepancy relies on the loss in the **NTF** attenuation due to finite gain of the operational amplifier employed which is not accounted for in the prediction formulae, but was already shown in figure 5.7. Lowering the **OSR** implies that the in-band quantization noise power increases becoming the predominant noise source, so its weigh over the **SNR** is higher like figure 5.1 illustrates.

A limitation in the mixer clock generation as inside the **FPGA** board used to implement the digital control blocks restricted the frequency coverage to around 17.4MHz. However, the switches used in the mixer assembling have their turn ‘on’ and turn ‘off’ times around 0.5ns, as described in section 5.1. If the phase shift introduced by the non-instantaneous switching time is accounted for, one should expect the 1dB compression point in the in-phase gain around 147 MHz which is below the switches 300MHz bandwidth, according to the switch component **SN74LVC2G66** data sheet [TEX2004]. If the appropriate high speed clock source and the digital control cells of the mixer are provided, the mixer should be able to reach much higher input frequencies, hence extending the prototype input frequency range.

From the result presented in this section, one of the most limiting performance factors was the quality of the digital control signals due to time jitter. Then to improve the **FAC SNR** a better quality primary digital clock must be used such as the one reported in [CHU2003] where the **RMS** value for the time jitter is less than 22 ps.

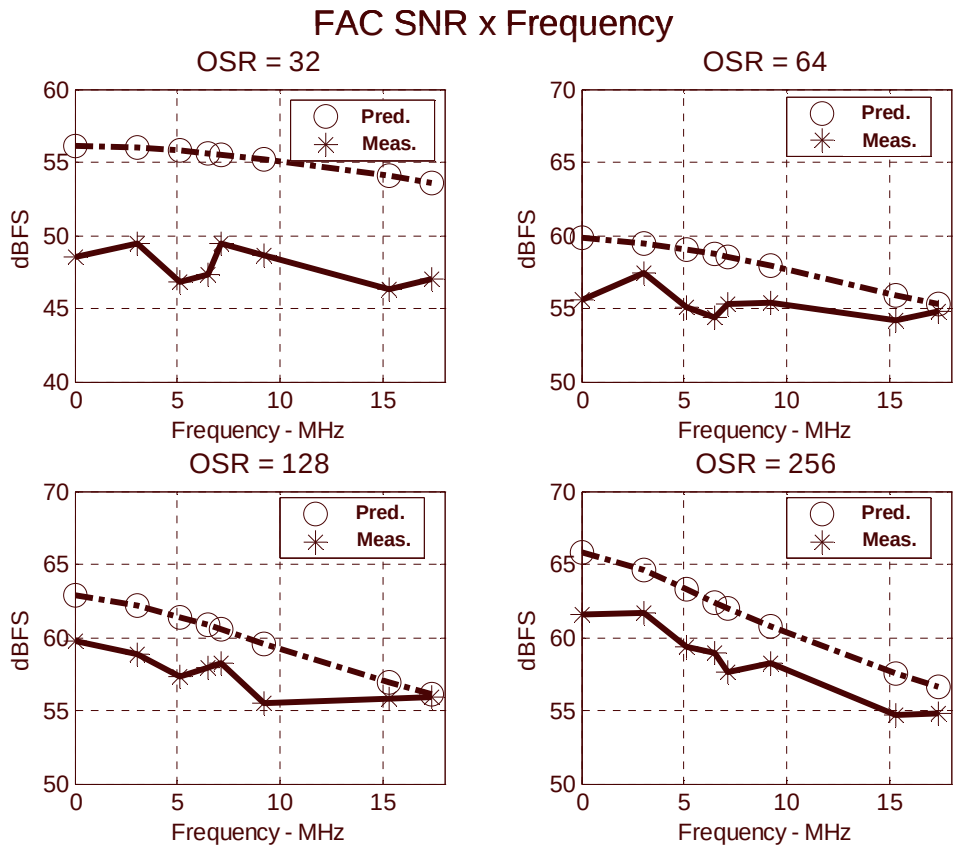


Figure 5.15: The prediction '**o**' and the prototype measurements '*****' **SNR FAC** frequency response for an **OSR** going from 32 up to 256.

The tested **FAC** prototype achieved good performance results when the presence of low quality digital clock generation in **FPGA** board is considered, such as:

1. frequency coverage from **DC** to 17.4 MHz (upper bound limit set by the **FPGA** clock source on board);
2. an **IMD3** better than 60 dBFS and **IIP3** greater than 15 dBFS;
3. an SFDR better than 69 dBFS for an OSR equal to 32 (**SBW** = 60 kHz);
4. a configurable **ENoB** through the decimator filter size (**OSR**) ranging from 8 bits to 12 bits.

Following this characterization process, some application mapping examples developed using the fixed analog cell were performed.

5.3 The Fixed Analog Cell Application Mapping

In this section, the proposed FAC and the reconfigurable digital infrastructure (**FPGA**) are used to implement some reconfigurable analog signal acquisition and processing functions. The application set developed shows the potential of the proposed interface architecture as a general reconfigurable platform.

5.3.1 The intrinsic analog-to-digital converter (multi-band) application

Analyzing the results shown in figure 5.15, it is straightforward to associate the **FAC** performance behavior with a multi-band data acquisition subsystem. Actually, it is only necessary to implement a base-band digital down-converter and a decimator stage and a reconfigurable multi-band **ADC** is ready to go. Moreover, this **ADC** is reconfigurable at application design time, not in the **FAC** design phase, since the digital block that implements the decimator to achieve the suitable **SNR** can be tailored during the end user application setup. The reconfigurability opens an additional tradeoff between bandwidth and resolution (**SNR**), as the user can balance bandwidth (OSR) and ENoB (SNR) to get the best relation for its application. From the frequency response point of view, an almost constant performance and large frequency coverage (from low to high frequencies) can be achieved provided the suitable low jitter mixer and ΔΣ modulator control signal as the evaluation process demonstrated above [FAB2003a and FAB2003b].

Indeed, two prototypes were built, the first one was a 1st order architecture implemented in low frequency with $\mathbf{f_P}$ around 1.5 kHz [FAB2003a, FAB2003b] and the other is the one described before. The first one, as a low frequency implementation, does not suffer from the timing issues previously pointed, since the relative jitter time to the sampling frequency and maximum mixer frequency tested is at least three decades below the ones reported in section 5.2.

To perform the analog-to-digital conversion a decimation filter is necessary. The $sinc^k$ filter is widely used and has an efficient architecture to execute the decimation process [VAZ2003]. Figure 5.16a shows the general form of a $sinc^k$ decimator and, an efficient topology for a 2nd order decimator is presented in figure 5.16b. This topology was used to implement the decimator.

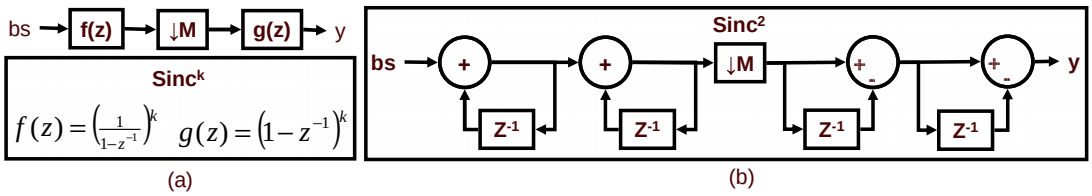


Figure 5.16: An efficient implementation of a *sinc* filter for decimation by a factor of **M**. In (a) k-order and in (b) a second order *sinc* filter.

The low frequency prototype was excited with tones near the frequencies shown in figure 5.17. The frequency of the mixer was adjusted to translate the applied signal to the processing frequency band with center equal to $\mathbf{f_P}$.

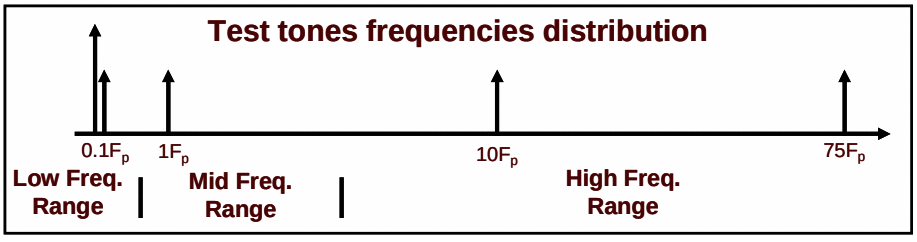


Figure 5.17.: Test tones frequencies distribution applied when the interface was configured as an analog-to-Digital Converter (Multi-Band).

Table 5.1 shows the logic element cost for each digital block that belongs to the control and signal processing part of the test prototype. The two prototypes tests showed that the **SFDR** is almost constant, around 49 dB, for the first order implementation with low frequency $\mathbf{f_P}$, and around 69 dB, for the second order high frequency implementation. The SFDR test was done making a low to high frequency sweep and employing a $sinc^2$ decimation window of 32 points. For each test tone applied, the signal-to-noise ratio (**SNR**) and the effective number of bits (**ENOB**) were measured for three different decimator sizes. The number of taps of the decimator has impact on **SNR** and output bandwidth (**BW**) from the signal processing perspective, and on the number of logic elements from the circuit implementation point of view. Table 5.2 summarizes the relationship between modulator order, bandwidth **BW**, **SNR** and the Altera™ **FPGA ACEX1K100** logic elements (LE) consumption.

Table 5.1: Synthesis results when using a $Sinc^1$ and $Sinc^2$ decimator with 128 taps.

Functional Block	$Sinc^1$ – LE	$Sinc^2$ – LE
Clock Generation and Mixer Control	35	35
ΣΔM Control	5	5
Decimator (128X)	56	138
Total	96	178

Table 5.2: The intrinsic ADC evaluation showing the SNR measurement as function of normalize frequency ($\mathbf{F_{norm}}$), modulator order **N**, bandwidth **BW** and the number of logic elements used, employing a *sinc2* decimator filter.

SNR [dB] / ENOB						
DECsize / BW / Total LE						
$\mathbf{F_{norm}}$	**64 / 4BW / 268 LE**		**128 / 2BW / 278 LE**		**256 / BW / 289 LE**	
	N = 1	**N = 2**	**N = 1**	**N = 2**	**N = 1**	**N = 2**
0.1	42 / 6.5	56 / 8	52 / 7.5	60 / 8.5	55 / 8	62 / 9
1	41 / 6.5	56 / 8	52 / 7.5	59 / 8.5	56 / 8	62 / 9
10	39 / 6	55 / 8	48 / 7	56 / 8	50 / 7.5	58 / 8.5
20	-	55 / 8	-	55 / 8	-	55 / 8
75	40 / 6	-	47 / 7	-	51 / 7.5	-

Analyzing each column of table 5.2 that represents a fixed BW and area cost (number of LEs used) for each input tone frequency, one can observe that overall performance (SNR and ENOB) remains almost constant. This shows that the proposed interface can deliver large frequency coverage with almost constant performance.

Inspecting Table 5.2 line wise one realizes that, for a fixed input tone frequency, one can balance **BW** against resolution. This also gives space for trading **OSR** (area cost) for power.

With a higher order analog resonator with better timing circuitry, this trade-off could be made even more aggressive [VAZ2003].

5.3.2 N channel adder

This interface can be used to add two analog input signal at their base-band. This operation is carried out at the bit stream (**BS**) level. There are two BS adding approaches that are shown in figure 5.18. The first approach requires a two-bit adder that results in a two-bit wide word stream. The requantizer low pass digital ΣΔ modulator that follows is needed to recover the result to **BS** format [DIA95].

The second approach interleaves the two input **BS**s generating directly an output BS at double frequency. This process is carried out by using a simple multiplexer. The extra hardware cost for adding two BS is in our prototype was only 1 logic element.

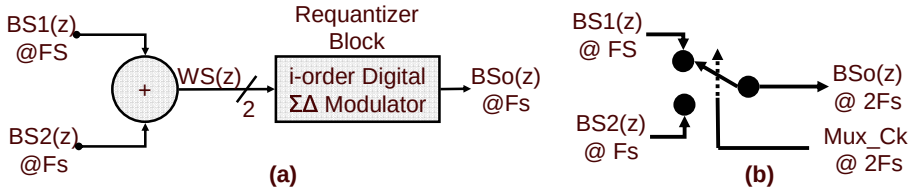


Figure 5.18: Structures for bit stream addition, in (a) a bit stream adder employing an adder cell, and in (b) a bit stream adder using a multiplexer with clock signal two times the bit stream sampling frequency.

For adding **N** analog channels, the interleaving procedure can be used and the resulting **BSo** sampling frequency will be N times the input sampling frequency.

5.3.3 Two channel analog multiplier

To build this application it is necessary to use two FACs. It is important to remark that two bit stream cannot be multiplied directly to avoid spreading the quantization noise signal band. Therefore, at least one channel must be decimated to filter out the quantization noise. This way, the multiplication of one bit stream signal and one word stream is carried out by a 2:1 multiplexer with the bit stream signal controlling the selection pin. The output of the multiplexer is a word stream that may be requantized. Figure 5.19 shows the analog multiplier application mapping over the proposed interface [FAB2004a].

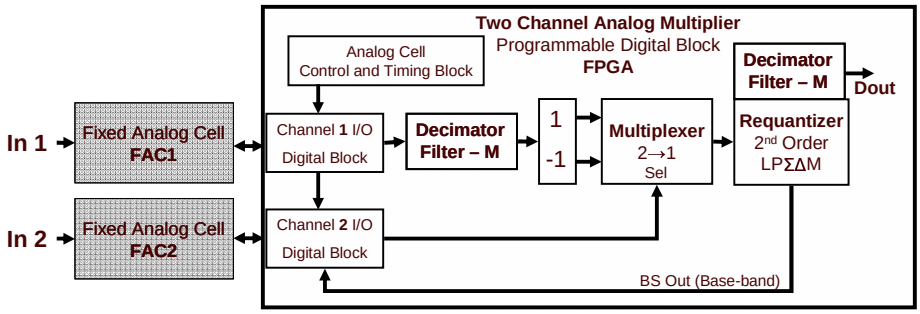


Figure 5.19: Two channel analog multiplier topology implemented in a FPGA device.

The realization cost for this application is shown in table 5.3. It is important to remark the reuse of many already designed digital blocks in the application development.

Table 5.3: Synthesis results for the two channel analog multiplier.

Functional Block	$Sinc^1$ – LE
Clock Generation, Mixer and ΣΔ Control, extra logic	61
Requantizer – 2nd Order LP digital ΣΔM (Input width:16 bits)	102
2 x Decimator (128X)	106
Total	269

In this section, some application capabilities of the proposed interface architecture using a fixed analog cell – **FAC** as a basic building were shown.

5.4 Discussion

This chapter presented the way the modeling approach developed in chapter 4 is used to make the design space exploration to settle the main topological parameters to start the circuitry design phase of the **FAC** blocks. The design space exploration was done using a typical target performance example. After the determination of the maximum allowable time jitter for the mixer and the ΔΣ modulator order, the minimum **OSR** and the quantizer number of bits, the circuitry design phase of the modulator was stressed.

The resulting designed prototype was tested and its performance was compared against the theoretical model developed. A few discrepancies were found from the measurements and the theoretical prediction. These errors were justified due to the fact that the model used neglects the excess time jitter in the ΔΣ modulator **DAC** control lines [TAO99a] and the finite gain product bandwidth of the operational amplifier employed, that could be overcome in an integrated circuit version by using a full custom approach [PAT2004].

As to the application of the FAC, the interface was used as a multi-band ADC with resolution and bandwidth configured at application design time. The frequency coverage was limited by the digital clock generation unit to 17.4 MHz, but the mixer components specification points to a 1 dB compression point around 140 MHz. It was also shown that the prototype could deliver at least 12.7 **ENoBs** for a 500 Hz input signal bandwidth, what makes the FAC eligible for applications in sensor data acquisition. The application for a multiplication of two analog signals was described and their synthesis results were shown, that it is a basic non-linear application.

The next chapter presents the final remarks, the conclusions and future work.

6 CONCLUSIONS AND FUTURE WORK

The IC technology scaling has created a fertile environment for integrating whole applications solutions in a single die as a system on a chip – SOC. In this SOC applications set, there are many solutions not only for purely digital applications like networking or high definition video processing units, but also examples of mixed-signal applications solutions such sensor signal conditioning. The requirements of these two SOCs classes are basically distinct from the input signal interfacing point of view, the purely digital may require high speed digital data interfacing to guarantee the necessary bandwidth for the data flow processing, on the other hand in the mixed-signal SOC scenario the demand could be an accurate analog signal interfacing capability. Then, the research focus of this thesis was to study and to propose a more general analog signal interface to be employed in the mixed-signal SOC applications with the objective of widening the applications mapping spectrum in the general purpose mixed-signal SOCs. So, the main objective of this research was to develop a mixed-signal interface that could deliver the following characteristics:

- performance almost constant from **DC** to high-frequencies for band limited signals, this way covering many applications;
- ability to balance between SNR and bandwidth according to the application constraints;
- capacity of realization of a wide variety of linear and non-linear applications;
- compatibility with digital control and programmability;
- CMOS technology compatible, to be in the industry main stream.

Therefore, a deep review over existing techniques for performing analog reconfiguration was done in the research work, because such a general purpose interface should incorporate some reconfiguration strategy to guarantee application and frequency coverage. From this review, we concluded that all analog reconfiguration solutions found had limitations, either in frequency coverage or in the range of applications mapping capability, or in both aspects.

The literature review has guided the research focus in the direction of an interface architecture with the analog building blocks fairly fixed and using frequency translation to reallocate the input signal to a fixed frequency for digital processing employing a reconfigurable digital hardware. This approach could avoid the penalties for doing reconfiguration in the analog building blocks as commonly employed in analog reconfigurable devices. The band-pass $\Delta\Sigma$ modulator was selected as the digitalization methodology of the reallocated input signal copy due to its robustness and compatibility

with digital **CMOS** processes. Additionally, the continuous time implementation strategy choice was made to simultaneously relax the anti-aliasing filter constrains, to eliminate sampling-and-hold devices and to employ the ΔΣ modulator loop filter as the image selection filter due to its intrinsic selectivity. The application mapping should be done using digital signal processing in the ΔΣ domain over the hardware available in mixed-signal SOC or over a digital reconfigurable infrastructure.

The methodology proposed for signal treatment based on frequency translation combined with the FAC architecture illustrated in figure 3.9 has some limitations. One of these constraints comprise the processing of a band-limited input signal buried in broadband spurious signals, since the original proposal of this thesis did not called for a band selection filter in front of the mixer. Having made this option, all spurious signals in the spectral positions defined by equation 3.5 are mapped back to the processing frequency f_P, with the corresponding translation gain (equation 3.4). This means that the usage of a switching mixer creates several spectral pass-through windows which can degrade the performance of the FAC in such specific situation, although the input desired signal is band-limited.

Another constraint aspect is related to the processing of very low amplitude input signals near DC. In this situation, it is necessary to use the VGA to amplify and adjust the input dynamic range, which in the presence of an input offset in the input stage can cause problems of saturation or reduction of the input signal dynamic range. This could be attenuated by adding some auto-zeroing function in the input stage. This limitation could not be compensated in the digital side because it is related to analog section dynamic range.

The high level performance prediction model of this interface was developed and used for design space exploration in an illustrative interface example. The resulting parameters from design space exploration were employed to design the experimental prototypes. Two prototypes were developed and tested in this research: one low order and low frequency and the one described in chapter 5. The low frequency prototype was used as a proof of concept built on proto-boards, operating at a low frequency ($\mathbf{f_P}$ = 1.5kHz), for which the effect of time jitter and the operational amplifier finite gain-product bandwidth are not limiting factors.

From the evaluation results of the low frequency prototype, the characteristic of frequency coverage with constant performance was achieved as table 5.2 shows. Actually, the prototype described in the previous chapter has its frequency coverage limitation and performance loss due to the bad quality of the digital main clock source, not by the **FAC** architecture itself. In the literature, examples are reported of digital PLLs or VCOs with an RMS time jitter that is at least one decade below the jitter of the clock generation circuitry used in the measured prototype [CHU2003]. Once a suitable reference time basis is provided by the digital control, the **FAC** performance could be extended almost a decade higher for the high frequency prototype case, as predicted by the performance model.

Both prototypes have shown the ability to trade off between bandwidth and equivalent number of bits – **ENoB** at the end user application design time that it is also synthesized in table 5.2. The base-band **SNR** and the bandwidth are directly related to the **OSR** and the required decimator filters which can be controlled and set at the application development time. This is certainly a strong advantage of the **FAC** concept

since it is not necessary to change the critical analog blocks of the **FAC** for adapting the resolution of an **ADC** application, for example.

From the application point of view, the interface was used to implement some basic linear and non-linear operations. The literature describes many other examples using the ΔΣ as the mixed-signal processing domain, as addressed in chapter 3.

Although high performance mixed signal processing can be achieved by the use of the proposed reconfigurable cell, it certainly has drawbacks. Besides the area penalty, the power dissipation is expected to be much higher than a dedicated analog circuit. Low frequency CMOS analog processing can be performed at very low biasing currents, even in the moderate-to-weak inversion regime. The proposed programmable architecture uses intensive signal processing developed at a high sampling rate. On the other hand, digital FPGAs are also much more power-hungry than their ASICs counterparts for the implementation of the same function, as hardware programmability comes at some power cost.

Thanks to the proposed analog prototyping technique, there is a great advantage**:** the application designer can reach the analog specifications of a new design very rapidly, since the constant performance, frequency-shifted digital signal processing part can be reprogrammed in the field. This way, the system designer can experiment different analog and mixed signal functions with digital FPGA programming, until reaching an optimum solution.

In our design architecture the fast analog prototyping is done in a design paradigm which is altogether different. Instead of changing the circuit topology, the designer can program the signal band translation. This has many useful implications, since the redesign or migration of the proposed configurable cell to other technologies is greatly simplified. Only the analog part of the modulator must be redesigned or targeted at the physical level to a new technology. This certainly simplifies the design process. The remaining analog processing circuits can be easily ported to a new technology, since they are simple large digital signal processing modules, and hence consolidated digital tools are available for their development.

6.1 Future Research Work

The experimental results have shown a deviation from the performance prediction model developed, because the FAC performance model neglects the attenuation loss of the loop filter in the ΔΣ modulator due to finite gain-product of the operational amplifiers and the jitter in the digital control signals. So, there is a research space for improvement on this model in such a way to incorporate those imperfections.

The FAC architecture and modeling employed the passive mixer topology and the continuous time ΔΣ modulator. We believe this research left an opportunity for investigation toward the potential performance achievement by exploiting other mixer and ΔΣ modulator topologies. An example could be the adjoined usage of a tuned active mixer and a mixed continuous and discrete time ΔΣ modulator, with the possible performance benefits regarding the input dynamic range and the **FAC SNR**.

The design and construction of a discrete prototype have validated the proposed interface architecture, so the next step is to integrate the whole fixed analog cell in chip to achieve the expected full benefits from integration.

It was clear from the prototype evaluation process that a critical point in achieving high performance at high frequencies is the clock generation unit. The investigation of high frequency dynamic range frequency synthesizers with time jitter to be used as the timing circuitry for both the mixer and the ΔΣ modulator is also an interesting field for research.

Since the ΔΣ domain is a true mixed-signal working space with a lot of high quality signal generation capabilities, we want to investigate some test techniques that could be applied to turn the FAC interface architecture testable and which resources, if any, should be necessary to incorporate to it for self-testability.

And the last but not the least, to extend the application mapping developed here to applications in the field of sensor conditioning (low frequency range) and IF signal acquisition and processing in the telecommunication field.

REFERENCE

[ADI2003] ANALOG DEVICES. Available at: <www.analog.com>. Visited on Dec. 2003.

[AGI2004] AGILENT INC. Available at: <www.agilent.com>. Visited on Dec. 2004.

[ALT2003] ALTERA INC. Available at: <www.altera.com>. Visited on Dec. 2003.

[ANA2003] ANADIGM INC. Availabel at: <www.anadigm.com>. Visited on Dec. 2003.

[BEN48] BENNETT, W.R. Spectra of quantized signals. **Bell System Technical Journal**, [S.l.], v. 27, p. 446-472, July 1948.

[BRA96] BRATT, A.; MACBETH, I. Design and Implementation of a FPAA. In: ACM/SIGDA INTERNATIONAL SYMPOSIUM ON FIELD-PROGRAMMABLE GATE ARRAYS, FPGA, 1996. **Proceedings…** [S.l.:s.n.], 1996. p. 88-93.

[BRA98] BRATT, A.; MACBETH, I. DPAD2 – A Field-Programmable Analog Array. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 67-90, Sept. 1998.

[CAT2002] CATUNDA, S.Y. et al. Programming an anlog signal conditioning circuit wihtout loss in measurement range. In: IEEE INSTRUMENTATION AND MEASUREMENT TECHNICAL CONFERENCE, IMTC, 2002. **Proceedings…** [S.l.:s.n.], 2002. p. 529-534.

[CHA96] CHANG, R. C. et al. Programmable Weight Building Blocks for Analog VLSI Neural Networks Processors. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 9, n. 3, p. 215-230, Apr. 1996.

[CHE99] CHERRY, J.; SNELGROVE, W. Clock jitter and quantizer metastability in continuous-time delta-sigma modulators. **IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing**, New York, v. 46, n. 6, p. 661-676, June 1999.

[CHO95] CHOW, P.; CHOW, P.; GULAK, P.G. A Field-Programmable Mixed-Analog-Digital Array. In: ACM/SIGDA INTERNATIONAL SYMPOSIUM ON FIELD-PROGRAMMABLE GATE ARRAYS, FPGA, 1995. **Proceedings…** [S.l.:s.n.], 1995. p. 104-109.

[CHU2003] CHUNG, C.C. , LEE, C.Y. An All-Digital Phase-Locked Loop for High-Speed Clock Generation. **IEEE Journal of Solid-State Circuits**, New York, v. 38, n. 2, p. 347-351, Feb. 2003.

[COR2002] CORTES, F. P. et al. A ΣΔ A/D Converter insensitive to SEU effects. In: THE EIGHTH IEEE INTERNATIONAL ON-LINE TESTING WORKSHOP, IOLTW, 2002. **Proceedings…** [S.l.:s.n.], 2002. p. 89-93.

[COR2003] CORTES, F.P.; FABRIS, E.E.; BAMPI, S. Analysis and design of amplifiers and comparators in CMOS 0.35 m technology. **Microelectronics Realabitly**, [S.l], v. 44, n. 4, p. 657-664, Apr. 2004.

[CYP2003] CYPRESS MICROSYSTEM. Available at: <www.cypress.com>. Visited on Dec. 2003.

[DAL2003] DALLAS SEMICONDUCTOR. Available at: <www.dalsemi.com>. Visited on Dec. 2003.

[DIA95] DIAS, V.F. Signal processing in the sigma-delta domain. **Microelectronics Journal**, [S.l.], v. 26, p. 543-562, 1995.

[DIC2000] DICK, C.; HARRIS, F. FPGA signal processing using sigma-delta modulation. **IEEE Signal Processing Magazine**, New York, n. 1, p. 20-35, Jan. 2000.

[DME98] D'MELLO, D.R.; GULAK, P.G. Design approaches to field-programmable analog integrated circuits. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 7–34, Sept. 1998.

[DOB2002] DOBOLI, A.; VEMURI, R. A functional specification notation for co-design of mixed analog-digital systems. In: DESIGN, AUTOMATION AND TEST IN EUROPE CONFERENCE AND EXHIBITION, DATE, 2002. **Proceedings…** Los Alamitos: IEEE Computer Society, 2002. p. 760-769.

[EMB98] EMBABI, S.H. et al. A current-mode based field-programmable analog array for signal processing aplications. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 125-142, Sept. 1998.

[FAB2003a] FABRIS, E.E.; CARRO, L.; BAMPI, S. An analog signal interface with constant performance for SOCs. In: The IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 2003. **Proceedings…** Los Alamitos: IEEE Circuits and Systems Society. 2003. v. 1, p. 773-776.

[FAB2003b] FABRIS, E.E.; CARRO, L.; BAMPI, S. An universal high-performance analog interface for signal processing SOCs. In: SYMPOSIUM ON INTEGRATED CIRCUITS AND SYSTEMS DESIGN, 16., 2003, São Paulo. **Proceedings…** Los Alamitos: IEEE Computer Society, 2003. p. 137-142.

[FAB2004a] FABRIS, E.E.; CARRO, L.; BAMPI, S. Analog Signal Processing Reconfiguration for Systems-on-Chip Using a Fixed Analog Cell Approach. In: INTERNATIONAL CONFERENCE ON FIELD-PROGRAMMABLE LOGIC AND ITS APPLICATIONS, FPL, 2004, Anthuerp, BE. **Proceedings…** [S.l]: Springer Verlag, 2004. p. 1136-1138.

[FAB2004b] FABRIS, E.E.; CARRO, L.; BAMPI, S. Modeling and Designing High Performance Analog Reconfigurable Circuits. In: SYMPOSIUM ON INTEGRATED CIRCUITS AND SYSTEMS DESIGN, 17., 2004, Porto de Galinhas, PE. **Proceedings…** New York: ACM Computer Society, 2004. p. 49-54.

[FAU97] FAURA, J. et al. A novel mixed signal programmable device with on-chip microprocessor. In: IEEE CUSTOM INTEGRATED CIRCUITS CONFERENCE, 1997. **Proceedings…** Los Alamitos: IEEE Solid State Circuit Society, 1997.

[GIN2002] GINÉS, A.J.; PERALÍAS, E.; RUEDA, A.; MADRID, N.M.; SEEPOLD, R. A. Mixed-signal design reuse methodology based on parametric behavioural models with non-ideal effects. In: DESIGN, AUTOMATION AND TEST IN EUROPE CONFERENCE AND EXHIBITION, DATE, 2002. **Proceedings…** Los Alamitos: IEEE Computer Society, 2002.

[GIR2003] GIRARDI, A.; BAMPI, S. LIT - An Automatic Layout Generation Tool for Trapezoidal Association of Transistors for Basic Analog Building Blocks. In: DESIGN, AUTOMATION AND TEST IN EUROPE CONFERENCE AND EXHIBITION, DATE, 2003, Munich. **Proceedings…** Los Alamitos: IEEE Computer Society, 2003. p. 1106-1107.

[HER96] HERNANDEZ, L. Frequency synthesis based on bandpass sigma-delta modulation. **IEE Electronic Letters**, New York, v. 32, n. 18, Aug. 1996.

[JON93] JOHNS, D.A.; LEWIS, D.M. Design and analysis of Delta-Sigma based IIR filters. **IEEE Transactions on Ciircuits and Systems II: Analog and Digital Signal Processing**, New York, v. 40, p. 233-240, Apr. 1993.

[KLE98] KLEIN, H.W. The EPAC Architecture: An Expert Cell Approach to Field Programmable Analog Devices. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 91–104, Sept. 1998.

[KRA2001] KRAVER, K.L. et al. A mixed signal sensor interface microinstrument. **Sensors and Actuators A: Physical**, [S.l.], v. 91, n. 3, p. 266-277, July 2001.

[KUB90] KUB, F. J. et al. Programmable Analog Vector-Matrix Multipliers. **IEEE Journal of Solid State Circuits**, New York, v.25, n. 2, p. 207-214, Feb. 1990.

[KUT98] KUTUK, H.; KANG, S. A Switched-Capacitor Approach to Field-Programmable Analog Array (FPAA) Design. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 51-66, Sept. 1998.

[LAT2003] LATTICE SEMICONDUCTOR CORP. Available at: <www.latticesemi.com>. Visited on Dec. 2003.

[LAT98] LATHI, B.P. **Modern Digital and Analog Communication Systems**. 3rd ed. New York: Oxford University Press, 1998.

[LEE92] LEE, E.; GULAK, P. G. Field-Progammable Analogue Array Based on MOSFET Transconductor. **IEE Eletronic Letters**, New York, v.28, n. 1, p. 28-29, Jan. 1992.

[LEE95] LEE, E.; GULAK, P.G. A Transconductor Based Field-Progammable Analog Array. In: IEEE SOLID-STATE CIRCUITS CONFERENCE, ISSCC, 1995. **Digest of Technical Papers**. Los Alamitos: IEEE Solid-State Circuits Society, 1995. p. 198-199.

[LEE98a] LEE, E.K.; HUI, W.L. A Novel Switched-Capacitor Based Field-Programmable Analog Array Architecture. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.] v. 17, n. 1/2, p. 35-50, Sept. 1998.

[LEE98b] LEE, T.H. **Design of CMOS Radio-Frequency Integrated Circuits**. New York: Cambridge University Press, 1998.

[LEV2002] LEVIN, P.L.; LUDWIG, R. Crossroads for Mixed-Signal Chips. **IEEE Spectrum**, New York, v. 39, n. 3, p. 38-43, Mar. 2002.

[LUA94] LU, A. K.; ROBERTS, G.; JOHNS, D. A high quality analog oscillator using oversampling D/A conversion techniques. **IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing**, New York, v. 41, n. 7, p. 437-444, July 1994.

[MAL2003] MALCOVATI, P. et al. Behavioral Modeling of Switched-Capacitor Sigma–Delta Modulators. **IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications**, New York, v. 50, n. 3, p. 352-364, Mar. 2003.

[MAL92] MALOBERTI, F. Non Conventional Signal Processing by the Use of Sigma- Delta Technique: A Tutorial Introduction. In: THE IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 1992. **Proceedings…** Los Alamitos: IEEE Circuits and Systems Society, 1992. v. 6, p. 2645-2648.

[MAR2002] MARTIN, K.W. et al. **Digital Integrated Circuit Design**. New York: Oxford University Press, 2000.

[MOH2003] MOHAMED, A.I. et al. A 60-dB 246-MHz CMOS Variable Gain Amplifier for Subsampling GSM Receivers. **IEEE Transactions on Very Large Scale Integration Systems**, New York, v. 11, n. 5, p. 835-838, Oct. 2003.

[NAT2003] NATIONAL SEMICONDUCTOR. Available at: <www.national.com>. Visited on Dec. 2003.

[NOR97] NORTHWORTHY, S.R.; SCHREIER, R.; TEMES, G.C. **Delta-Sigma Data Converters:** Theory, Design and Simulation. Piscataway: IEEE Press, 1997.

[OLE90] O'LEARY, P.; MALOBERTI, F. Bit-stream adder for oversampling coded data. **IEE Electronics Letters**, New York, v. 26, n. 20, p. 1708-1709, Sept. 1990.

[ONS2003] ON SEMICONDUCTOR. Available at: <www.on.com>. Visited on Dec. 2003.

[ORT2003] ORTMUNNS, M. et al. Influence of Finite Integrator Gain Bandwidth on Continuous-Time Sigma Delta Modulators. In: THE IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 2003. **Proceedings…** Los Alamitos: IEEE Circuits and Systems Society, 2003. v. 1, p. 925-928.

[PAN2002] PANKIEWICZ, B. et al. A Field Programmable Analog Array for CMOSnContinuous-Time OTA-C Filter Applications. **IEEE Journal of Solid-State Circuits**, New York, v. 37, n. 2, p. 125-136, Feb. 2002.

[PAT2004] PATON, S. et al. A 70-mW 300-MHz CMOS Continuous-Time ΔΣ ADC With 15-MHz Bandwidth and 11 Bits of Resolution. **IEEE Journal of Solid-State Circuits**, New York, v. 39, n. 7, p. 1056-1063, July 2004.

[PIE98a] PIERZCHALA, E.; PERKOWSKI, M.A. A High-Frequency Field Programmable Analog Array (FPAA) Part 1: Design. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 143–156, Sept. 1998.

[PIE98b] PIERZCHALA, E.; PERKOWSKI, M.A. A High-Frequency Field Programmable Analog Array (FPAA) Part 2: Applications. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l.], v. 17, n. 1/2, p. 157-169, Sept. 1998.

[PLA2001] VAN DER PLAS, G. et al. AMGIE — A Synthesis Environment for CMOS Analog Integrated Circuits. **IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems**, New York, v. 20, n. 9, p. 1037-1058, Sept. 2001.

[PRE98] PREMONT, C. et al. A Current Conveyor Based Field-Programmable Analog Array. **Analog Integrated Circuits and Signal Processing: An International Journal**, [S.l], v. 17, n. 1/2, p. 105-124, Sept. 1998.

[PRO95] PROAKIS, J.G.; MANOLAKIS, D.G.; MANOLAKIS, D. **Digital Signal Processing:** Principles, Algorithms and Applications. 3rd ed. [S.l.]: Pearson Education, 1995.

[RAZ96] RAZAVI, B. A Study of Phase Noise in CMOS Oscillators. **IEEE Journal of Solid-State Circuits**, New York, v. 31, n. 3, p. 331-343, Mar. 1996.

[RAZ98] RAZAVI, B. **RF Microeletronics**. Upper Saddle River: Prentice Hall, 1998.

[RIL94] RILEY, T.; COPELAND, M. A simplified continuous phase modulator technique. **IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing**, New York, v. 41, n. 5, p. 321-328, May 1994.

[SAH2002] SAHU, B.; DUTTA, A.K. Automatic Synthesis of CMOS Operational Amplifiers: A Fuzzy Optimization Approach. In: THE ASIA AND SOUTH PACIFIC DESIGN AUTOMATION CONFERENCE, ASP-DAC, 7., 2002. **Proceedings…** Los Alamitos: IEEE Computer Society, 2002. p. 366-371.

[SHE86] SHEINGOLD, D.H. (Ed.). **Analog-Digital Conversion Handbook**. Englewood Cliffs: PTR Prentice Hall, 1986.

[SHI2002] SHIBATA, H.; FUJII, S.M. Automated Design of Analog Circuits Using Cell-based Structure. In: THE IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 2002. **Proceedings…** Los Alamitos: IEEE Cicuits and Systems Society, 2002. v. 2, p. 53-56.

[SHI90] SHINAGAWA, M. et al. Jitter analysis of high speed sampling systems. **IEEE Journal of Solid-State Circuits**, New York, v. 25, n. 1, p. 220-224, Jan. 1990.

[SID2003] SIDSA S.A. Available at: <www.sidsa.com>. Visited on Dec. 2003.

[SIV88] SIVILOTTI, M. A Dynamically Configurable Architecture for Prototyping Analog Circuits. In: MIT VLSI CONFERENCE, 1988. **Proceedings…** [S.l.:n.s.], 1988. p. 237-258.

[SOM2002] SOMMER, R. et al. From System Specification To Layout: Seamless Top-Down Design Methods for Analog and Mixed-Signal Applications. In: DESIGN, AUTOMATION AND TEST IN EUROPE CONFERENCE AND EXHIBITION, DATE, 2002. **Proceedings…** Los Alamitos: IEEE Computer Society, 2002.

[TAO99a] TAO, H.; KHOURY, J.M. Analysis of Timing Jitter in Bandpass Sigma–Delta Modulators. **IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing**, New York, v. 46, n. 8, p. 991-1001, Aug. 1999.

[TAO99b] TAO, H. et. al. A 400-Ms/s Frequency Translating Bandpass ΣΔ Modulator for A/D Conversion in Digital Radio. **IEEE Journal of Solid-State Circuits**, New York, v. 34, n. 12, p. 1741-1752, Dec. 1999.

[TEX2004] TEXAS INSTRUMENTS. Available at: <www.ti.com>. Visited on July 2004.

[TOU2002] TOUMAZOU, C. et al. (Ed.). **Trade-offs in Analog Circuit Design – The Designer's Companion**. Dordrecht, The Netherlands: Kluwer Academic Publishers, 2002.

[VAZ2003] RODRÍGUEZ-VÁZQUEZ, A. et al. (Ed.). **CMOS Telecom Data Converter**. Boston: Kluwer Academic Publishers, 2003.

[VEI97] VEILLETTI, B. R.; ROBERTS, G. Amplitude modulated signal generation using third-order Delta-Sigma oscillator. In: THE IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, ISCAS, 1997. **Proceedings…** Los Alamitos: IEEE Circuits and Systems Society, 1997. v. 1, p. 397-400.

[WAN2002] WANG, H.; VRUDHULA, S.B. Behavioral synthesis of field programmable analog array circuits. **ACM Transactions On Design Automation of Electronic Systems**, New York, v. 7, n. 4, p. 563-604, Oct. 2002.

[WON90] WONG, P.; GRAY, R. FIR filter with sigmal-delta modulation encoding. **IEEE Transactions on Acustic, Speech and Signal Processing**, New York, v. ASSP-38, n. 6, p. 979-990, June 1990.

[XIC2003] XICOR INC. Availabel at: <www.xicor.com>. Visited on Dec. 2003.

APPENDIX A THE THEORETICAL MODELS FOR THE PROTOTYPE DESIGN

In this appendix, the mathematical procedure used to make the transformation from the discrete time model to continuous time transfer function and the corresponding determination each feedback **DAC** gains.

It is also shown the influence of the gain and gain product bandwidth of an operational amplifier in the frequency response of integrator and resonator.

A.1 Discrete to Continuous time transformation

Actually the objective of this section is to show the theoretical procedure behind equation 4.21 that is reproduced again in A.1. This simple equation states the equivalence between the sampled version of a **CT** system given by the transfer function $\mathbf{R_{DAC}(s)xH_{CTR}(s)}$ to a **DT** system with **Z** transform equal to $\mathbf{H_R(z)}$. In others words, it says that the time response at the sampling instants of the **CT** system must match the response to the **DT** system response at the same time.

$$Z^{-1}\{H_R(z)\}_{t=nT} = L^{-1}\{R_{DAC}(s)H_{CTR}(s)\}_{t=nT} \qquad \text{(A.1)}$$

This way, the exact **DAC** pulse shape that excites the **CT** resonator must be known for making the equivalence stated by equation A.1. Therefore, figure A.1 shows the selected continuous time band-pass ΔΣ modulator topology that we want to match to the discrete time prototype. Indeed, the objective is to determine the gain of each **DAC** feedback path that is represented by $\mathbf{H1_{HRZ}}$, $\mathbf{H2_{HRZ}}$, $\mathbf{R2_{RZ}}$ and $\mathbf{R2_{RZ}}$ in figure A.1, in such that a way the signal **y(t)** matches **DT** prototype **y[n]** in each sampling instant.

One way to calculate these constants is finding the **DT** transfer function equivalent from the **DAC** up to the quantizer input that actually is the loop filter transfer function, and equating the zeros of the **DT** response of the **CT** resonator to the **DT** response of the prototype. In figure A.1, the loop transfer function corresponds to the transfer function from point $\mathbf{A_{DT}}$ to point $\mathbf{B_{DT}}$. These zeros are functions of the desired gains.

The **DT** starting point is the low-pass loop filter as described in section 4 and as equation I.2 shows.

$$Hlp := \frac{-2z + 1}{(z - 1)^2} \qquad \text{(A.2)}$$

Applying the low-pass to band-pass filter transformation given by (A.3) where **a** is equal to **–cos(** Ω **)** and accounting that for this design the selected Ω is **π/4** what makes the band-pass **DT** loop filter equal to equation A.4.

$$LP2BP := -\frac{z\,(z + a)}{a\,z + 1} \tag{A.3}$$

$$Hzbp := \frac{2\,z^2 + 1}{(z^2 + 1)^2} \tag{A.4}$$

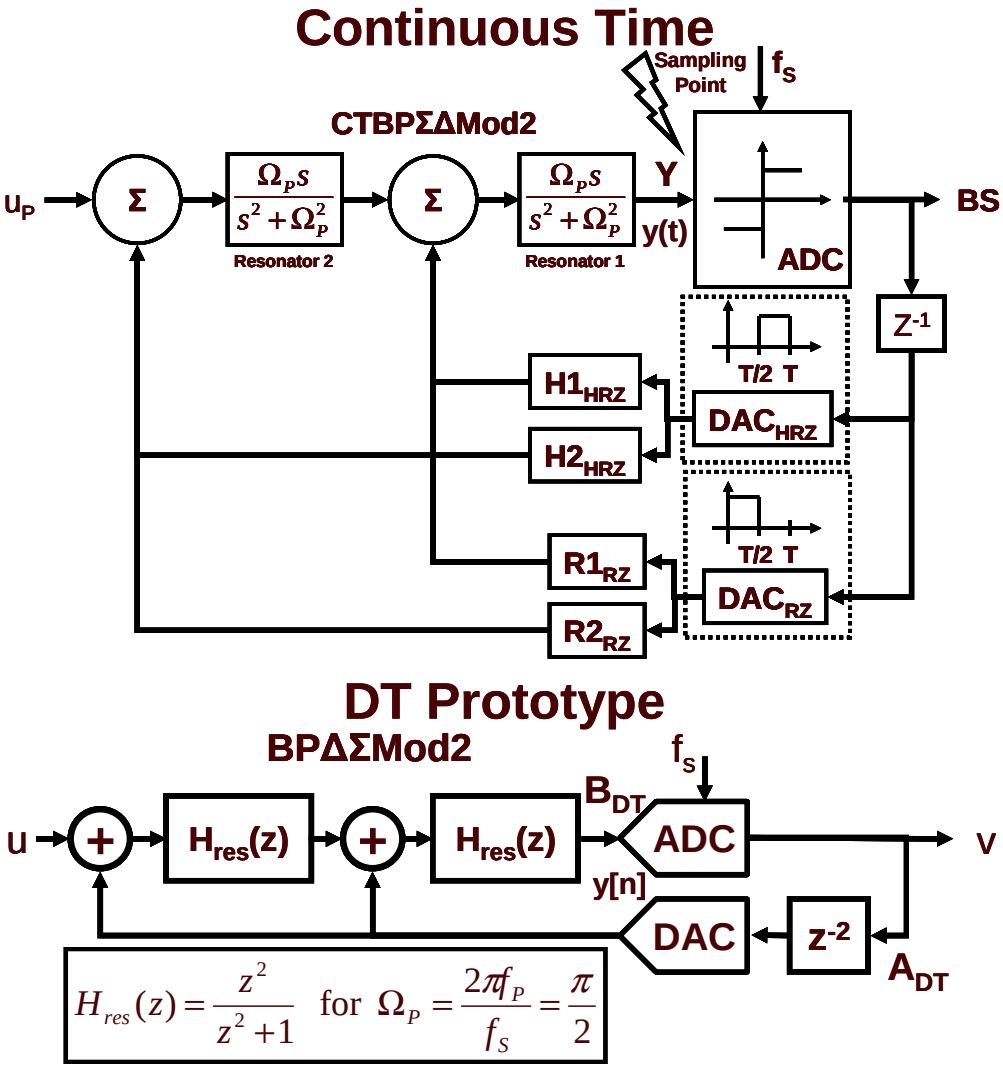


Figure A.1: The continuous time band-pass ΔΣ modulator architecture.

Since the **DAC** pulses have transitions inside the sampling period, the *Z transform* can not be used directly to carry out **CT** to **DT** transformation, actually this transformation must use the so called *Modified Z transform* that incorporates a delay in its definition. This delay is less than a sampling period, as shown by equation A.5

$$Zm_{f(t),\,m} := \sum_{k=0}^{\infty}\left(\frac{f(k\,T - (1 - m)\,T)}{z^k}\right) \quad \text{where } 0 < m < 1 \tag{A.5}$$

Table A.1 shows the *Laplace* (**L{f(t)}**), the **z** (**Z{f(t)}**) and the **modified-z** (**Z_m{f(t)}**) transforms of some of the time functions that are used in this development.

Table A.1: The *Laplace* (**L{f(t)}**), the **z** (**Z{f(t)}**) and the **modified-z** (**Z_m{f(t)}**) transforms table of some time domain functions.

f(t)	L{f(t)}	Z{f(t)}	Z_m{f(t)}
$u_{-1}(t)$	$\frac{1}{s}$	$\frac{z}{z-1}$	$\frac{1}{z-1}$
e^{-pt}	$\frac{1}{s+p}$	$\frac{z}{z-e^{-p}}$	$\frac{e^{-p\cdot m}}{z-e^{-p}}$
$\cos(\Omega t)$	$\frac{s}{s^2+\Omega^2}$	$\frac{z\cdot[z-\cos(\Omega)]}{z^2-2\cos(\Omega)z+1}$	$\frac{z\cdot\cos(\Omega\cdot m)-\cos[\Omega\cdot(1-m)]}{z^2-2\cos(\Omega)z+1}$
$\sin(\Omega t)$	$\frac{\Omega}{s^2+\Omega^2}$	$\frac{z\cdot\sin(\Omega)}{z^2-2\cos(\Omega)z+1}$	$\frac{z\cdot\sin(\Omega\cdot m)+\sin[\Omega\cdot(1-m)]}{z^2-2\cos(\Omega)z+1}$
$t\cdot\cos(\Omega t)$	$\frac{s^2-\Omega^2}{(s^2+\Omega^2)^2}$	Eq. A.6 with m=1	Eq. A.6
$t\cdot\sin(\Omega t)$	$\frac{2\Omega s}{(s^2+\Omega^2)^2}$	Eq. A.7 with m=1	Eq. A.7

$$\frac{z\cdot m\cdot\sin(\Omega\cdot m)+(m-1)\cdot\sin[\Omega\cdot(1-m)]}{z^2-2\cos(\Omega)z+1}-\frac{2z\sin(\Omega)\{z\sin(m\Omega)+\sin[\Omega\cdot(1-m)]\}}{(z^2-2\cos(\Omega)z+1)^2} \quad (A.6)$$

$$\frac{z\cdot m\cdot\cos(\Omega\cdot m)+(1+m)\cdot\cos[\Omega\cdot(1-m)]}{z^2-2\cos(\Omega)z+1}+\frac{2z\sin(\Omega)\{z\cos(m\Omega)-\cos[\Omega\cdot(1-m)]\}}{(z^2-2\cos(\Omega)z+1)^2} \quad (A.7)$$

The ΔΣ modulator structure is composed of two cascaded resonators where each resonator has the normalized transfer function **Hr(s)** as equation I.8 shows, **A** being the transfer function gain. The transfer function **Hr(s)** is normalized to the sampling frequency $\mathbf{f_S}$. Therefore, **CT** loop filter transfer function is divided in two branches, one from point $\mathbf{A1_{CT}}$ to point $\mathbf{B_{CT}}$ that it is equal to **Hr(s)** and the other one from point $\mathbf{A2_{CT}}$ to point $\mathbf{B_{CT}}$ that is equal to $\mathbf{Hr(s)^2}$.

$$Hr := \frac{A\,s}{s^2+\frac{1}{4}\pi^2} \quad (A.8)$$

The step response of each the **CT** loop filters nets linking one or two resonators is calculated using equation A.8. The step response of **Net1** and **Net2** (figure A.1) are equal to equations A.9 and A.10 respectively.

$$hn1 := \frac{2\,A\sin\left(\frac{1}{2}\pi t\right)}{\pi} \quad (A.9)$$

$$hn2 := \frac{A^2\,t\sin\left(\frac{1}{2}\pi t\right)}{\pi} \quad (A.10)$$

From the step response determined above and using a normalized sampling frequency to 1, the Z-transform of signals **hn1** and **hn2** are given by equations A.11 and A.12.

$$Hzn1 := \frac{2\,A\,z}{\pi\,(z^2+1)} \tag{A.11}$$

$$Hzn2 := \frac{A^2\,z\,(z^2-1)}{\pi\,(z^2+1)^2} \tag{A.12}$$

The last two equations state the **CT** to **DT** equivalency of the resonators step response for a step being applied in t = 0.

From figure A.1, either $\mathbf{DAC_{HRZ}}$ or $\mathbf{DAC_{RZ}}$ contribute to the time responses of **Net1** and **Net2**, so the full time response of the **DAC**s pulses at point $\mathbf{B_{CT}}$ is analyzed by accounting the effect each **DAC** pulse separately and adding those effects to get the complete response.

The $\mathbf{DAC_{RZ}}$ pulse has a transition in t = 0 to one and it remains at the same level until half the sampling when it returns to zero, so this pulse is modeled as two steps one positive at t = 0 and one negative at t = 0.5. The analysis of $\mathbf{DAC_{RZ}}$ contribution is divided into two effects one through **Net1** and other through **Net2**. The effect of $\mathbf{DAC_{RZ}}$ in **Net1** is computed using the result of equation A.11 and for the step at t=0.5 the modified transform of the resonator step (equation A.9) is used from table A.1 with m=0.5, this composition is shown in equation A.13.

$$YzR1 := \frac{2\,A\,z}{\pi\,(z^2+1)} - \frac{2\,A\left(\frac{1}{2}z\sqrt{2}+\frac{1}{2}\sqrt{2}\right)}{\pi\,(z^2+1)} \tag{A.13}$$

The $\mathbf{DAC_{RZ}}$ contribution in **Net2** employs exactly the same procedure executed for **Net1**, the only difference is that equations A.10 and A.12 are used instead of equations A.9 and A.11, respectively, what results in equation A.14 as $\mathbf{DAC_{RZ}}$ contribution in **Net2**.

$$YzR2 := \frac{A^2\,z\,(z^2-1)}{\pi\,(z^2+1)^2} - \frac{A^2\left(-\frac{-\frac{1}{4}z\sqrt{2}+\frac{1}{4}\sqrt{2}}{z^2+1}+\frac{2\left(\frac{1}{2}z\sqrt{2}-\frac{1}{2}\sqrt{2}\right)z}{(z^2+1)^2}\right)}{\pi} \tag{A.14}$$

The next step is to evaluate the contribution of the $\mathbf{DAC_{HRZ}}$ using the same steps employed for the $\mathbf{DAC_{RZ}}$ influence. At this time, the **DAC** pulse starts at t = 0.5 and lasts at t = 1, hence this pulse is divided in two steps one from zero to one at t = 0.5 and from zero to minus one at t = 1. Actually, the second step at t = 1 is simply the negative of the step response at t = 0 delayed by one sample or equations A.11 and A.12 multiplied by a minus unity delay or $\mathbf{-z^{-1}}$. Therefore, the $\mathbf{DAC_{HRZ}}$ contributions in **Net1** and **Net2** are expressed by equations A.16 and A.17 respectively.

$$YzH1 := -\frac{2\,A}{\pi\,(z^2+1)} + \frac{2\,A\left(\frac{1}{2}z\sqrt{2}+\frac{1}{2}\sqrt{2}\right)}{\pi\,(z^2+1)} \tag{A.16}$$

$$YzH2 := -\frac{A^2 (z^2 - 1)}{\pi (z^2 + 1)^2} + \frac{A^2 \left(-\frac{-\frac{1}{4} z \sqrt{2} + \frac{1}{4}\sqrt{2}}{z^2 + 1} + \frac{2\left(\frac{1}{2} z \sqrt{2} - \frac{1}{2}\sqrt{2}\right) z}{(z^2 + 1)^2} \right)}{\pi} \tag{A.17}$$

The whole response is computed using the individual effect and multiplying them by the respective **DAC** gains in the path as shown previously (figure A.1). The complete discrete time response at point $\mathbf{A_{CT}}$ is shown by equation A.18, where the constants $\mathbf{H1_{HRZ}}$, $\mathbf{H2_{HRZ}}$, $\mathbf{R2_{RZ}}$ and $\mathbf{R2_{RZ}}$ are the gains of each DACs paths, and the terms **YzR1**, **YzR2**, **YzH1** and **YzH2** are the **DAC** pulses responses stated by equation A.13 through A.17, respectively**.**

$$Y(z) = H2_{HRZ} \cdot YzH2 + H1_{HRZ} \cdot YzH1 + R2_{RZ} \cdot YzR2 + R1_{RZ} \cdot YzR1 \tag{A.18}$$

Actually, equation A.18 is the loop filter transfer function and its poles are all over the unity circle at $\mathbf{f_S/4}$ exactly the same way of A.4 (**DT** loop filter prototype) has. Hence, for matching A.4 and A.8 it necessary to find the gain constants in such a way that to equal the zeros of A.18 to the zeros of A.4, results in linear set of equations whose solution is shown in equation A.19 and the respective floating point values given by A.20.

$$CoeffSols := \left\{ R1 = -\frac{5}{128}\sqrt{2} + \frac{1}{64},\ H1 = \frac{7}{64} + \frac{7}{128}\sqrt{2},\ H2 = \frac{2 + \sqrt{2}}{16\,\pi},\ R2 = -\frac{\sqrt{2}}{16\,\pi} \right\} \tag{A.19}$$

CoeffSols = [-.4501581579, -.6338834762, 1.086777930, 2.987436867] (A.20)

The determination of the gain coefficients enables to go to the next phase in the

A.2 The Influence of the Operational Amplifier GBW in the Resonator Transfer Function

Figure A.2a shows the classical integrator circuit employing an operational amplifier with gain $\mathbf{A_{op}}$, a resistor with resistance $\mathbf{R_i}$ and a capacitor with capacitance $\mathbf{C_i}$, the integrator transfer function is written as equation A.21.

$$Hi := -\frac{Aop}{1 + (1 + Aop)\, C\, Ri\, s} \tag{A.21}$$

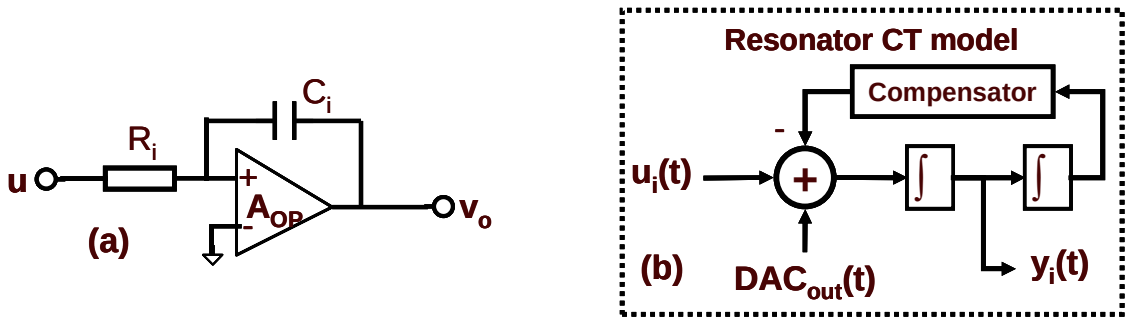


Figure A.2: (a) The active-**RC** integrator circuit and (b) the continuous time resonator architecture.

Using expression A.21 and the resonator topology shown in figure A.2, the resonator transfer function assuming the same gain for both integrators is given by expression A.22.

$$Hr\_ideal := -\frac{(1+(1+Aop)\,C\,Ri\,s)\,Aop}{(1+2\,Aop+Aop^2)\,s^2\,Ri^2\,C^2+(2\,Aop+2)\,s\,Ri\,C+Aop^2+1} \tag{A.22}$$

From expression A.22, the resonant frequency ω_0 is calculated and shown in equation A.23. Analyzing this result, a small frequency shift from ideal value $1/(\mathbf{R_iC_i})$ is observed and this deviation is smaller as the amplifier gain at ω_0 augments.

$$\omega 0 := \frac{\sqrt{Aop^2+1}}{Ri\,C\,(1+Aop)} \tag{A.23}$$

The merit factor of the resonator $\mathbf{Q_0}$ is calculated from expression A.22 and its valus is expressed by equation A.24. The corresponding gain $\mathbf{G_0}$ at the ω_0 frequency is shown by the formulae A.25. Both the gain and the merit factor are close to $\mathbf{A_{op}/2}$, for $\mathbf{A_{op}} >> 1$.

$$Q0 := \frac{1}{2}\sqrt{Aop^2+1} \tag{A.24}$$

$$G0 := \frac{\sqrt{2+2\,Aop+Aop^2}\,Aop}{2\sqrt{1+2\,Aop^2+2\,Aop}} \tag{A.25}$$

Considering the operational amplifier with a transfer function given by equation A.28, where $\mathbf{A_{dc}}$ is low frequency gain (**DC**), **p** is dominant pole and **GBW** is ain-product bandwidth of the operational amplifier, the equation A.27 expressed the transfer function of an integrator implemented with this amplifier.

$$A_{op}(s) = \frac{A_{dc}\cdot p}{s+p} \quad \text{where} \quad p = 2\pi\frac{GBW}{A_{dc}} \tag{A.26}$$

$$Hi\_real := -\frac{Adc\,p}{C\,Ri\,s^2+((p+Adc\,p)\,Ri\,C+1)\,s+p} \tag{A.27}$$

Equations A.27 shows that the finite amplifier **GBW** creates at least one additional pole at high frequency. Assuming an amplifier with the **DC** gain $\mathbf{A_{dc}}$ and the dominant pole are equal to 3300 and p = 0.07, actually this data corresponds to the operational amplifier used in the implementation described in chapter 5 but normalized to $\mathbf{R_iC_i = 1}$. This amplifier has an GBW around 200MHz for a gain over 10. Therefore, using expression A.27, figure A.2b and assuming that the summing node is also implemented with one extra amplifier, the resonator transfer function is shown in equation A.28.

$$H_r = -\frac{231.0000000\,(s+232.0696984)\,(s+115.5699999)\,(s+0.0003016335201)}{(s+233.0524961)\,(s+231.0699978)\,(s+115.6039991)\,(s^2-0.01649298154\,s+0.9900013961)} \tag{A.28}$$

Equation I.28 shows that when the 1st order amplifier model is considered the resonator loop become instable. So, during the prototype design phase a lead-lag compensator was introduced in the resonator loop to minimize this effect as figure I2.b shows.

APPENDIX B THE FIXED ANALOG CELL PROTOTYPE DESIGN AND TESTING

This appendix is divided into two sections the first one describes the **FAC** simulation model and the design steps involved in the prototype development and, the second section addresses the prototype assembling, initial tests and the prototype rework.

B.1 The FAC Simulation Model

The simulation model for the proposed interface was built using the Simulink™ suite of the Matlab™ mathematical tool using a hierarchical structure. Figure B.1 shows the top level of the simulation model that is composed by the input signal generation and the 2nd order continuous time ΔΣ modulator that produces the output bit-stream for analysis. A normalized sampling frequency equal to **1** is employed.

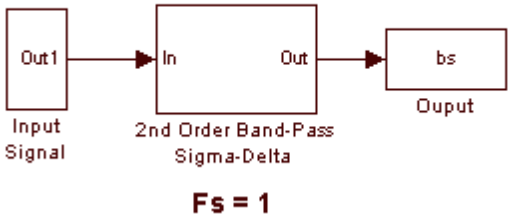


Figure B.1: The FAC Simulink™ model.

Figure B.2 shows the signal generation structure that has the mixer and its control signal and two tone signal generators to create compound signals. The mixer was modeled as a switch that is controlled by a discrete time pulse generator. The generators and the mixer frequencies are independently set by a Matlab™ initialization script file.

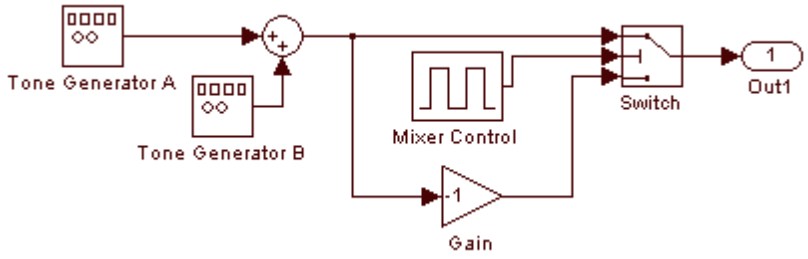


Figure B.2: The input signal generation model.

The 2nd order **CTBPΔΣMod** block diagram is shown in figure B.3 that was divided into three separate functions. The set composed by the sample and hold and the quantizer is modeled as a track and hold comparator and the discrete pulse generator sets sampling cadency. The sample and hold input signal is the output of the noise shaper filter and the quantizer output is the discrete time bit-stream. The bit-stream is one sample delayed and is directed to the **DAC** input to generate the two **CT** output pulses (**RZ** and **HRZ**) that they match the theoretical model shown in appendix A. The noise shaper block implements the **CT** loop filter.

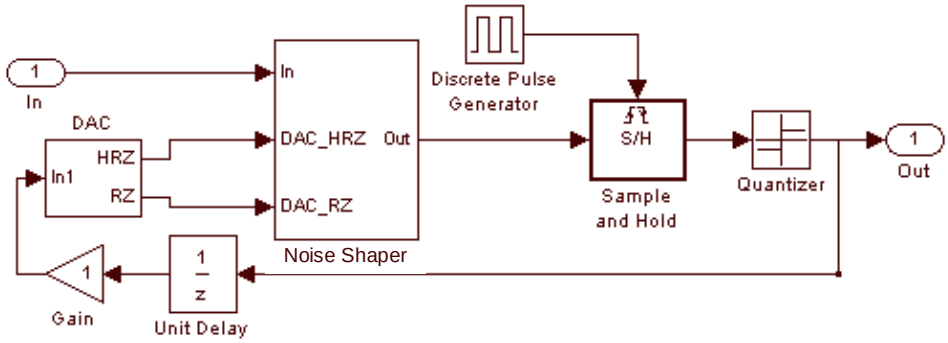


Figure B.3: The 2nd order band-pass continuous time model.

To implement the **DAC** pulse as defined by the topology choice depicted in figure A.1, the **DAC** uses two switches to create the return-to-zero – **RZ** and the half-delayed-return-to-zero – **HRZ** pulse shapes. The output of each switch commutes between the output bit-stream values (+1 or -1) and zero (the reference level), the control of the **DAC** switches is two **DT** pulse generators at the sampling frequency with duty cycle of 50%. The **HRZ** control is an 180° phase shifted version of the **RZ** control.

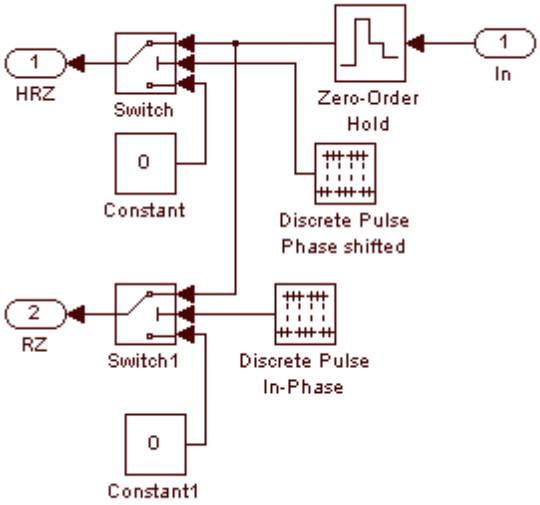


Figure B.4: The **DAC** with the **RZ** and **HRZ** pulse shapes.

Figure B.5 shows the **CT** model of the 2nd order band-pass loop filter. The gains of each **DAC** pulse to the resonators input are those calculated as shown in appendix A.

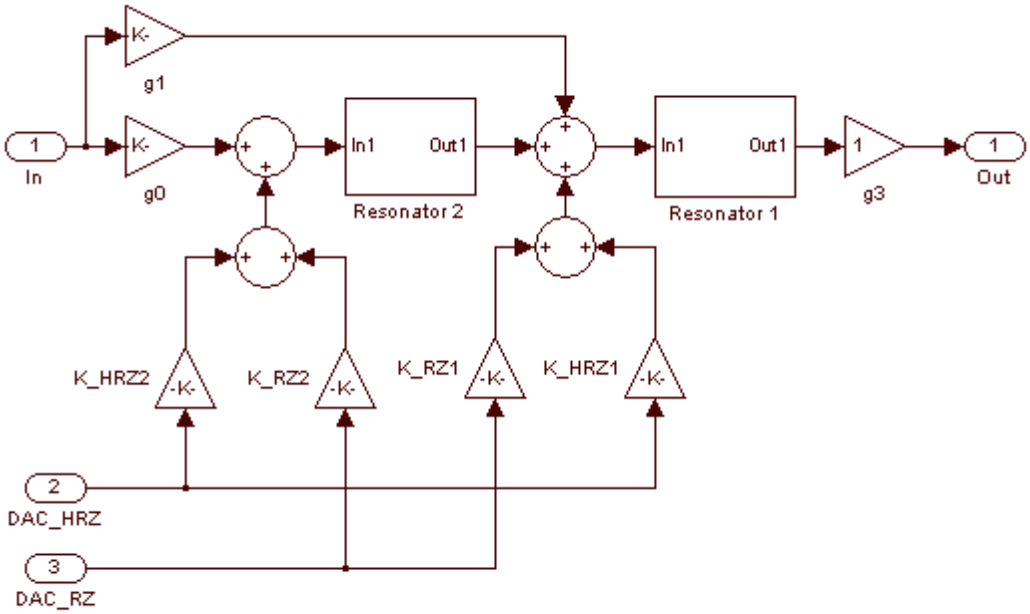


Figure B.5: The noise shaper loop filter model.

Each loop filter resonator has the classical state variable topology, as shown in figure B.6. The model permits the usage of different time constants for each resonator and the corresponding integrators. This feature is employed in the simulations of the impedance scaled variant. It was used initially the Simulink **CT** integrator in the model since it permits to set a saturation value either for the positive or the negative limit what permits the analysis of saturation effects in the performance of the modulator. The integrator gain is individually set by the gain block in front of it. The four integrators

outputs are sampled and collected during the simulation to allow the corresponding dynamic range analysis.

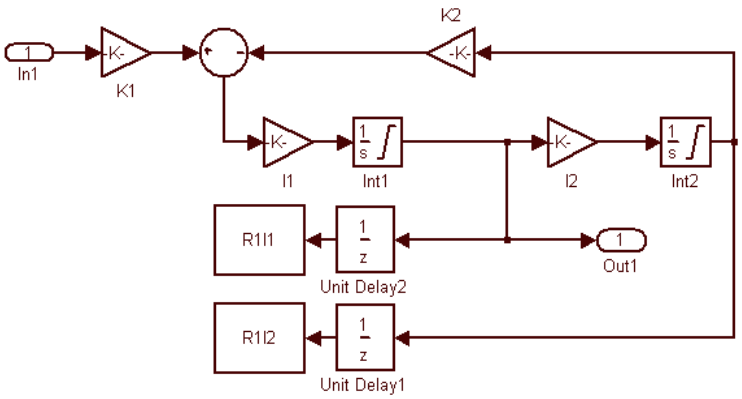


Figure B.6: The resonator Simulink™ simulation model.

The set of blocks described creates the simulation model for an ideal 2nd order CTBPΔΣMod. Figure B.7 shows a simulation result using the presented model with the parameters determined in appendix A.

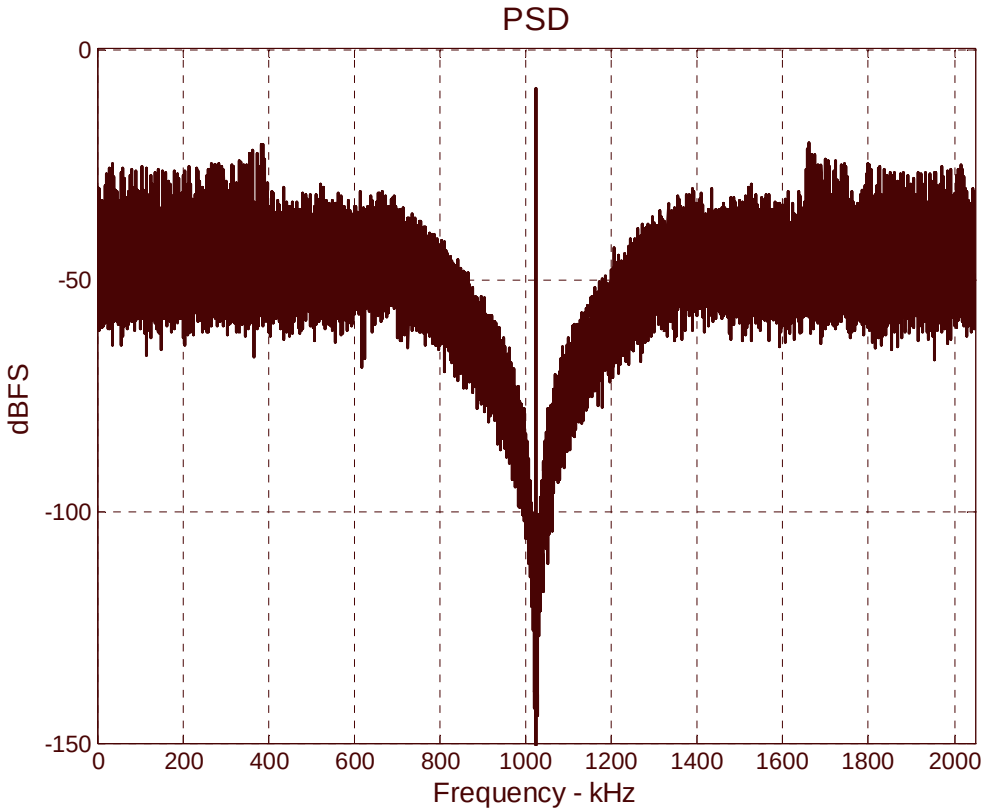


Figure B.7: The power spectral density of an ideal continuous time 2nd order band-pass ΔΣ modulator excited by an input signal near the $\mathbf{f_s/4}$ central frequency using the developed simulation model.

The ideal simulation model was used to validate the **CT** to **DT** transformation as shown in chapter 5. This ideal model was also employed to evaluate the individual output dynamic range of the four integrators to perform the impedance scaling and set the same output **DR** for all integrators.

The model presented was improved by accounting for the real world integrator imperfections, so the ideal integrators inside each resonator were substituted by an active **RC** integrator using operational amplifiers. Figure B.8 presents model for this new integrator and it is the classical implementation of inverter integrators with one operational amplifier inside of a feedback loop, actually. Additionally, the operational amplifier model was built using the 1st order approach that accounts for the **DC** gain **Aop** and the dominant pole **p_op** of the amplifier, this model is also shown in figure B.8. This more realistic approach was also used in the chapter 5 to predict the performance loss due to the amplifier finite gain-product bandwidth – **GBW** that is equal to the **DC** gain **Aop** times the dominant pole **p_op** divided by 2π.

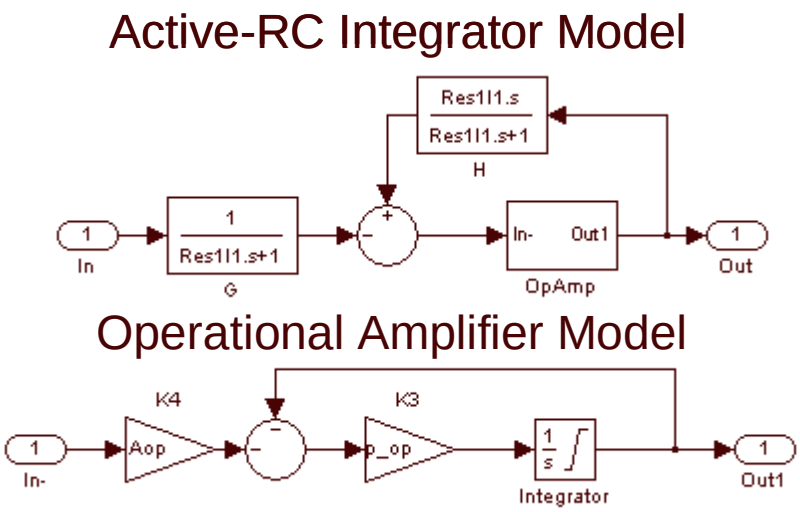


Figure B.8: The active-RC integrator model using an operational amplifier with finite gain-product bandwidth.

In the next section, the ΔΣ modulator model will be used to aid in the prototype component selection task.

B.2 The FAC prototype design and initial tests

Chapter 5 described the overall steps to reach the prototype assembling and evaluation. This section addresses some of the underneath tasks executed during the prototype design phase and debugging to achieve the results analyzed in the chapter 5.

B.2.1 The prototype design phase

The design space exploration was the first step to define the main constraints of the **FAC** mixer and ΔΣ modulator blocks. After the definition of the maximum allowable time jitter in the mixer and the order, the quantizer number of bits and the minimum **OSR** of the ΔΣ modulator, the architectures of those two blocks were defined and it has already been addressed in chapter 5. Following the architecture choice, the next phase is the physical design of each block.

The mixer physical designed was quite simple since it would be implemented as a balanced passive mixer, its design actually is the selection of set of switches with the appropriate turn ‘on’ and ‘off’ times, the ‘on’ state resistance and bandwidth that it was already addressed in chapter 5.

Certainly, the ΔΣ modulator physical design was more difficult and trickier than the mixer design. After the architecture definition and the feedback **DAC** gains determination, the ΔΣMod circuitry design was started. This way, the next physical design steps were the following:

1. the active components selection;
2. the impedance and the reference scaling for input and output dynamic range adjustment;
3. the passive component calculation;
4. the prototype printed-circuit-board – **PCB** development.

The operational amplifier selected was the OPA3690 and OPA2690 from Texas Instruments [TEX2004] and they have three and two identical operational amplifiers respectively with a product-gain-bandwidth – **GBW** equal to approximately 220 MHz. The OPA3690 was used in the resonator circuitry and the OPA2690 in the **DAC** pulse generation.

Looking for the determination of the necessary signal dynamic of each integrator, the ideal **CT** ΔΣ modulator simulation model was used to collect the output signal values of the four integrators along the time, the collected values were analyzed and corresponding maximum signal excursion of each integrator was evaluated. Figure B.9 illustrates the signal dynamic range of each integrator output without impedance scaling. The signal excursion and activity (**RMS**) value is higher in the resonator 2 than resonator 1, the resonator 2 is the one closer to the input signal as shown by figure II.5. The first integrator has more activity (RMS) than the second integrator in both resonators.

The simulation results shown in figure B.9 were used to make the integrators impedance scaling in such a way to achieve almost the same output dynamic range in all of them without modifying the loop filter frequency response, this procedure has the objective to use all the available integrator amplifier **DR** to improve system dynamic performance. After the scaling adjustment, the outputs of all integrator are swinging almost the same levels as figure B.10 shows.

From figure B.10, the output excursion of each integrator is between -4.5 and +4.5 approximately, but the selected operational amplifier has an output swing stuck between -3.5V and +3.5V for a power supply voltage of ±5V. Therefore, the reference level of the **DAC** was also compressed by changing its gain, as shown in the figure B.3, from 1.0 to 0.4 and, it was also scaled the gain **g0** (figure B.5) proportionally, so the ΔΣ modulator input **DR** was preserved. The reference level scaling limited the integrators **DR** between -2.5V and 2.5V, it was left a safe margin to avoid saturation problems.

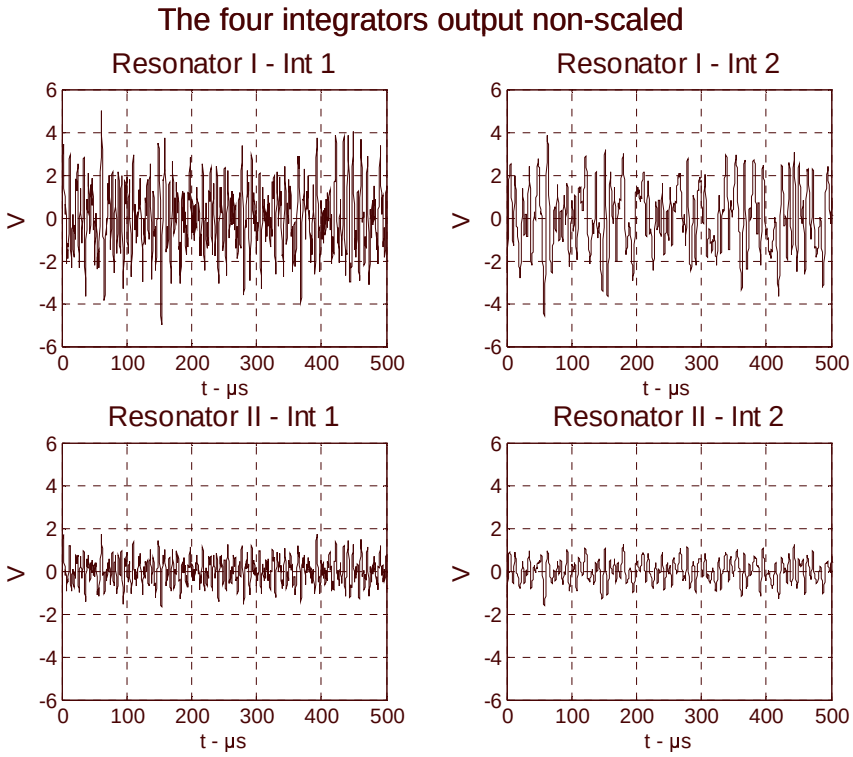


Figure B.9: The output signal dynamic range of each integrator in the 2nd order CT band-pass ΔΣ modulator, without impedance scaling.

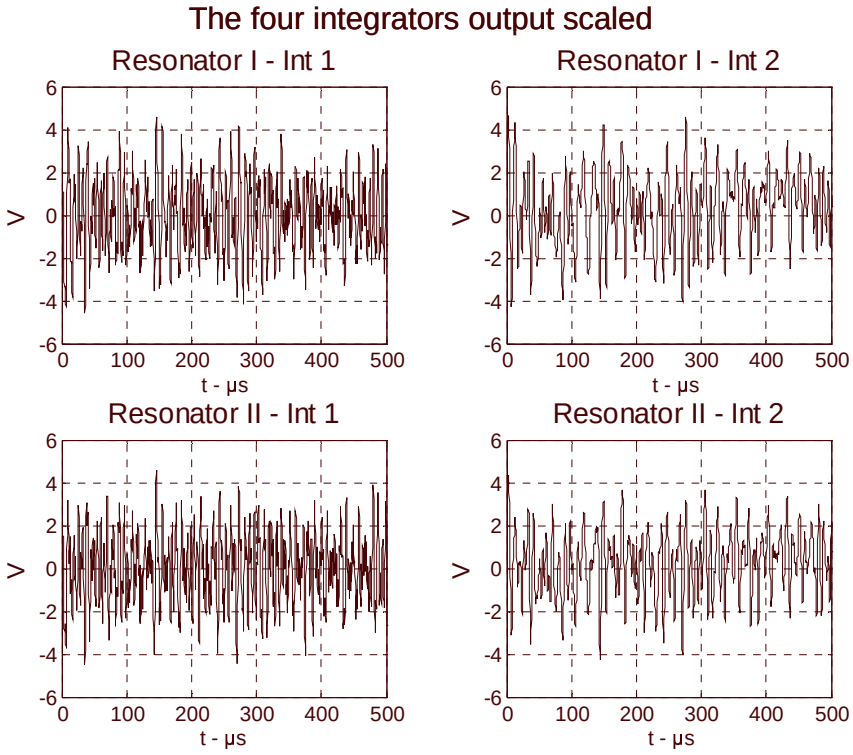


Figure B.10: The output signal dynamic range of each integrator in the 2nd order CT band-pass ΔΣ modulator with impedance scaling.

After the scaling process, the passive component values were calculated. With all components values specified, the prototype printed circuit board has been designed and also sent to fabrication. As soon as the fabricated **PCB** returned, the prototype was assembled as figure B.11 shows. Actually, three identical prototypes to be used in the evaluation tests were assembled.

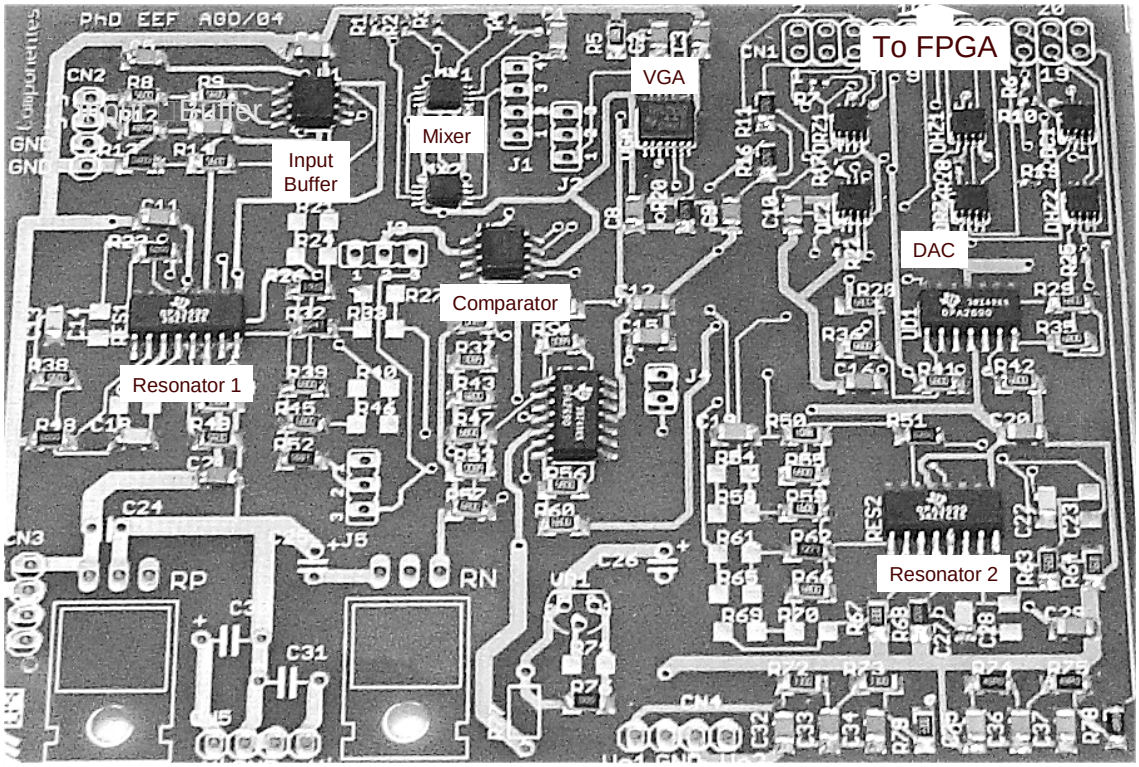


Figure B.11: The assembled prototype photography identifying the main blocks location in the printed circuit board.

The initial tests and the debugging are addressed in next section.

B.2.2 The FAC prototype debugging and evaluation

This section addresses the initial tests and the debugging strategy used to become the prototype fully operational. The start point was the visual inspection of the assembled **PCB** to search for missing components or bad connections and fixed the found problems.

After passing successfully on the visual inspection phase, the next step was to power up the whole system with **FPGA** attached and to program the **ACEX1K100** on the **FPGA** board with the test code. The test code implemented the basic controlling features of the **FAC** resources and it was used to validate the **DAC** and the mixer signals. In this test phase, several faulty bonds (soldering problems) were found and corrected. So, the prototype was operational and ready to start collecting the bit-stream data for analysis.

One of the initial tests was done by exciting **FAC** prototype with a 9.217 MHz tone signal near the input full scale amplitude (around $1V_p$) and the resulting bit-stream was collected and analyzed. The figure B.12 shows the **PSD** of this measurement. Bad news, the **FAC PSD** did not match the expected shape and, also, there are noise peaking near **DC** and $\mathbf{f_S/2}$, as shown by the same figure. Moreover, the noise attenuation near the

processing frequency was too low even though the amplifier finite gain product was accounted for. Furthermore, if the measured **PSD** is expanded around the processing frequency, many distortion tone peaks are observed as figure B.13 depicts. We went back to the theoretical analysis and made some modification into the simulation model in such a way to introduce imperfections for trying to mimic the measurement, so we could return directly in the circuit to analyze the specific cause. The following list addresses the possible cause-effect relationships of the initially identified issues:

1. the strong **DC** level – an offset voltage at the comparator input could cause this effect;
2. the noise peak at $\mathbf{f_S/2}$ – power supply noise coupling of sampling frequency harmonics;
3. the noise attenuation degradation and distortion – operational amplifier saturation, noise at the **DAC** reference voltage and time jitter in the ΔΣ modulator control lines.

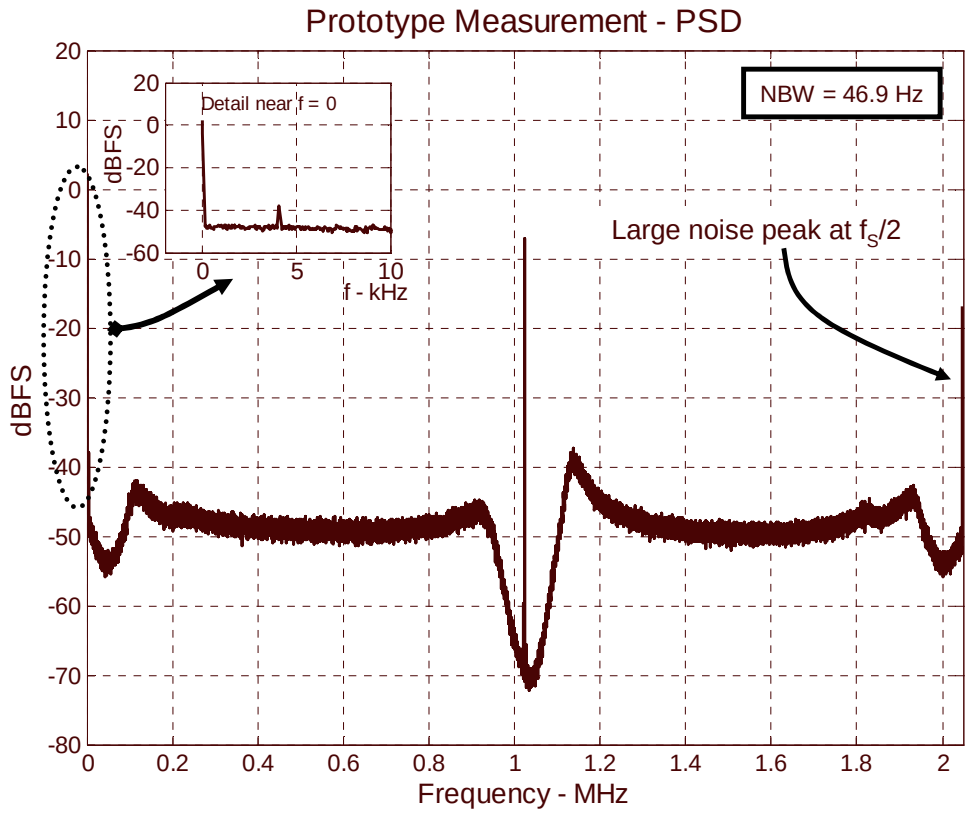


Figure B.12: The FAC measured PSD in the initial test phase.

With the possible causes list defined, a signal 'sniffing' process was started to confirm the performance degradation causes listed before. But, the only possible identified problem was a bouncing in the **DAC** reference signal lines that it was promptly corrected. Actually, we could not identify the real problem because the oscilloscope used in the debugging processing, by the time of this test, was a digitizing scope from HP with only 100MSa/s. This scope, due to its 'relatively small bandwidth', had hidden some high frequency oscillations (around 230 MHz) spread all over the board. This spurious oscillation was identified when the 1 GHz Infiinium™ **DSO** was

received by the lab and it was employed in the debugging and evaluation process, as mentioned in chapter 5 in the instrumentation setup.

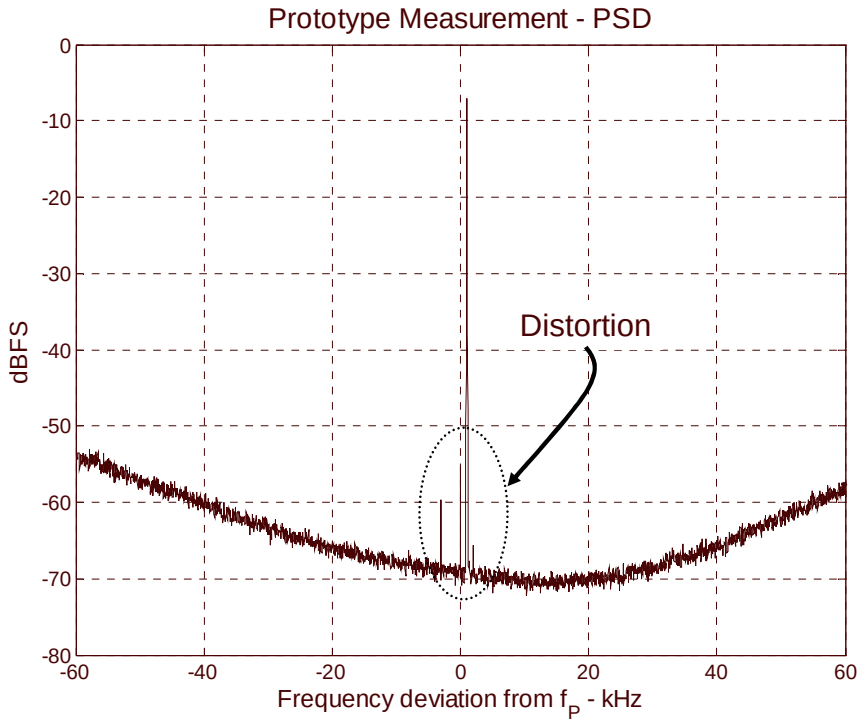


Figure B.13: The FAC measured PSD expanded around the processing frequency $\mathbf{f_P}$ in the initial test phase.

The high frequency oscillation was major cause of the problems initially observed. It was necessary to find the oscillation source, so we went back to the theoretical analysis and account for the full frequency response of the operational amplifier in the frequency resonator, using the Maple™ and the Matlab™ tools. It was found that a phase shift of 180^o in the resonator transfer function loop gain occurred near the measured oscillating frequency (230 MHz), but the calculated gain margin was around 40 dB which is sufficiently high to avoid oscillations. But, at 230 MHz the **PCB** wires do not behave as an ideal short circuits, indeed they have an inductance associated with them and, also, magnetic and capacitive coupling to other lines. Therefore, the stray inductances and capacitances could be the cause for the measured behavior. This ways, the critical feedback paths were made by flying wire over the **PCB** and, the long connection paths were resistively terminated. After, the 'cut' and 'rewire' process the prototype became fully operational without any oscillation and its behavior as it was characterized in chapter 5. This process took about two months of work.

Figure B.14 shows the photo of fully operational reworked **FAC** prototype and the attached FPGA board employed in the characterization process. It is interesting to remark that **PCB** design were done using the well known rules regarding analog blocks layout to avoid noise coupling like converging to one point power and signal grounds, the usage of ground plane in critical locations and the appropriate power supply decoupling, but this was not sufficient to avoid the mentioned issues.

After a long hard working on prototype measurements and analysis the prototype could be used in the validation task of the focus of this thesis.

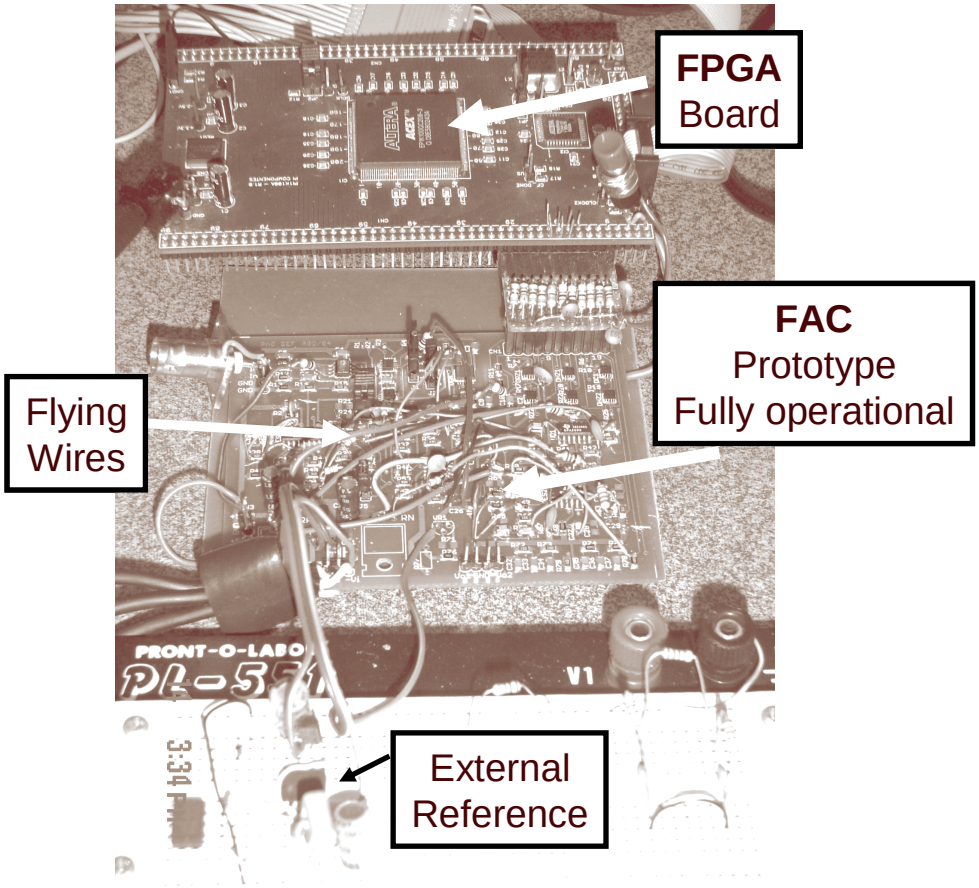


Figure B.14: The debug prototype photography identifying the main blocks location in the printed circuit board.

APPENDIX C UMA INTERFACE MODULAR E DIGITALMENTE PROGRAMÁVEL BASEADA EM MODULADORES SIGMA-DELTA PASSA-BANDA PARA SISTEMAS EM CHIP DE SINAIS MISTOS

O capítulo 1 desta tese apresenta a motivação e coloca o cenário em que esta pesquisa está inserida. A principal motivação deste trabalho vem do contínuo avanço da tecnologia de integração de dispositivos CMOS em silício que tem possibilitado ao projetista a inclusão de um número cada vez maior de blocos funcionais no mesmo circuito integrado. Este avanço crescente vem sendo impulsionado pela rápida expansão das necessidades mercadológicas por sistemas em chip (**SOC**). Este incremento da demanda por circuitos integrados cada vez mais complexos cria, por outro lado, a necessidade não só por ferramentas de apoio ao projeto automatizado, mas também para a prototipação de tais sistemas. Dentro do leque de aplicações de sinais mistos (analógicos e digitais), podem ser encontrados exemplos que envolvem tanto o processamento linear ou não linear de sinais, como também a necessidade de cobertura de uma faixa ampla de freqüências, indo das baixas às altas freqüências. Logo, não só as ferramentas de CAD necessitam suportar esse ambiente de projeto de sinais mistos, mas também as plataformas utilizadas para o mapeamento dessas aplicações devem ter o condizente suporte. Assim, a incorporação de algum grau de programação das funcionalidades analógicas mostra-se necessária nos atuais sistemas digitais reconfiguráveis ou SOCs de propósito geral. Nesse sentido, o foco desta tese é a descrição e validação de uma arquitetura de interface para processamento de sinais analógicos para SOC de sinais mistos, que possa ser reconfigurada no tempo de desenvolvimento da aplicação alvo. As características principais desta interface são de:

- processar sinais de banda limitada;
- apresentar cobertura de reposta em freqüência ampla indo de DC até HF;
- permitir o mapeamento tanto de aplicações lineares como não-lineares;
- permitir o balanço entre resolução (SNR) e largura de banda no momento da configuração da aplicação;
- permitir facilmente o controle da configuração por programação digital da mesma;
- apresentar compatibilidade com tecnologia digital CMOS convencional.

Em uma primeira abordagem, a arquitetura desta interface poderia envolver o emprego de um bloco reconfigurável analógico do tipo FPAA ou FPMA, associado a uma infra-estrutura de conversão analógico-digital e digital-analógico. Esta associação

poderia prover o interfaceamento analógico flexível como desejado. Objetivando analisar a viabilidade deste tipo de solução e suas limitações, foi conduzida uma revisão sobre dispositivos reconfiguráveis analógicos e de sinais mistos.

As técnicas de reconfiguração analógica, reportadas tanto na literatura como na indústria, são analisadas no capítulo 2. As arquiteturas de dispositivos analógicos reconfiguráveis existentes são classificadas em dois grandes grupos quanto a sua funcionalidade e ilustrada pela figura 2.2. Um grupo é composto pelos dispositivos que promovem reconfiguração paramétrica e outro é composto por soluções que premiam a reconfiguração estrutural. Os primeiros englobam as soluções cuja funcionalidade do dispositivo não pode ser alterada, tendo como exemplos comuns os filtros analógicos reconfiguráveis, os amplificadores de ganho variável e os potenciômetros digitais. Já o segundo grupo agrega as soluções que possibilitam a alteração da função desempenhada pelo dispositivo, podendo o FPAA ou FPMA ser configurado ora como um filtro ora como um conversor analógico-digital ou um oscilador, por exemplo. Dentro dos dispositivos programáveis estruturais encontram-se exemplos de arquiteturas tanto empregando abordagem de tempo contínuo como de tempo discreto (amostrados). A soluções em tempo discreto empregam a técnica de capacitores chaveados principalmente.

A tabela 2.1 sintetiza as principais características de cada uma das topologias encontradas quanto à tecnologia de integração utilizada, a cobertura de resposta em freqüência e conjunto de funções que podem ser mapeadas no dispositivo programável. Fica evidente deste processo de análise que não existe solução envolvendo dispositivos analógicos reconfiguráveis que atenda aos requisitos enumerados anteriormente, como mostra a figura 2.7. Isto ocorre pois, as arquiteturas que possibilitam cobertura de aplicações não contemplam a resposta em freqüência. Já, por outro lado, as topologias voltadas para uso em alta freqüência não são adequadas para o uso em aplicações em baixas freqüências, além do fato de sua tecnologia alvo ser do tipo bipolar, portanto não compatível com tecnologia CMOS. Após a análise das soluções para reconfiguração analógica, conclui-se que justamente os dispositivos adicionais inseridos no caminho do sinal ou da polarização dos **CABs** promovem a degradação do sinal, redução da faixa dinâmica e a própria resposta em freqüência. Para viabilizar uma ampla cobertura no lado das aplicações alvo, é necessária uma alta granularidade, que por sua vez impacta negativamente nos parâmetros gerais de desempenho. Conclui-se que existe um espaço para pesquisa que busque minimizar a problemas inerentes aos FPAAs e FPMAs.

O foco desta tese é a descrição e validação de uma arquitetura de interface para processamento de sinais analógicos para SOC de sinais mistos, cujo objetivo é:

- ampliar a faixa de utilização em freqüência;
- ampliar a cobertura de aplicações;
- criar um bloco voltado ao reuso para projetos envolvendo SOCs de sinais mistos.

A abordagem proposta apresenta a possibilidade de cobertura de uma larga faixa de freqüências com performance praticamente constante, associada a uma estrutura digital de programação. A premissa é usar uma célula analógica fixa e promover a configuração da aplicação no domínio digital, levando a uma arquitetura de interface de sinais mistos [FAB2003a, FAB2003b]. O emprego de um bloco analógico fixo busca eliminar a perda inerente de performance decorrente da própria estrutura de programação em circuitos reconfiguráveis analógicos. A promoção da programação no

domínio digital abre espaço para usos da vasta gama de ferramentas disponíveis para o projeto em alto nível de abstração, simulação e síntese automática para implementar a aplicação alvo com excelente predição do desempenho final.

No capítulo 3, o conceito de processamento geral de sinais empregando translação em freqüência (mixagem) da entrada analógica é apresentado. O uso da realocação em freqüência busca criar uma cópia do sinal em uma posição fixa do espectro (banda de processamento) seguida do mapeamento para o domínio digital para tratamento do sinal segundo os requisitos da aplicação. A vantagem inerente desta abordagem é o processamento homogêneo do sinal independentemente de sua freqüência central.

Pelo fluxo de processamento do sinal proposto, há a necessidade da conversão do sinal para um formato de representação digital através de um conversor analógico-digital (ADC) e de um filtro seletor de imagem após o mixer. Existem duas abordagens possíveis para etapa de conversão analógico-digital, sendo uma baseada em conversores operando no conceito da freqüência de Nyquist e outra utilizando técnicas de sobre-amostragem. Dentro das técnicas de conversão empregando sobre-amostragem os moduladores $\Delta\Sigma$ são os mais representativos, principalmente devido sua robustez, compatibilidade com tecnologia CMOS convencional e natureza intrínseca de representação de sinais mistas [DIA95]. A opção pelo emprego de um modulador $\Delta\Sigma$ passa-banda contínuo é devida às seguintes razões:

- eliminar a necessidade de amplificador amostrador-retentor (*sample-hold*);
- utilizar o filtro de conformação de ruído para efetuar a seleção da imagem do sinal e atuar como filtro *anti-aliasing*;
- utilizar o domínio $\Delta\Sigma$ como base para tratamento dos sinais;
- robustez e compatibilidade CMOS.

A estrutura de processamento proposta possibilita o emprego de um bloco analógico constante, e também, um processamento uniforme de sinais de entrada indo de DC até altas freqüências. A aplicação é configurada no domínio $\Sigma\Delta$ onde a performance pode ser predita de acordo com as especificações alvo. A arquitetura da *fixed analog cell* – FAC é apresentada na figura 3.9. O conceito do emprego da FAC como um bloco básico para construção de uma interface multi-canal é mostrado na figura 3.10.

No capítulo 4 é construído o modelo teórico de desempenho da FAC. O objetivo da construção deste modelo é criar um mecanismo para exploração do espaço de projeto em alto nível e extrair as especificações de cada bloco constituinte da FAC tendo como alvo a performance desejada no conjunto de aplicações alvo.

A estratégia para modelamento da performance global da FAC foi segmentá-la em dois blocos estanques, um compreendendo o mixer e outro agregando o modulador $\Delta\Sigma$.

O desempenho do mixer foi modelado empregando a arquitetura de um mixer passivo, onde a principal fonte de degradação é a incerteza temporal (*time jitter*) presente nas linhas de controle do bloco. O modelo foi construído utilizando expansão em série de Fourier dos sinais de controle do mixer (oscilador local). O *time jitter* foi modelado como um desvio de fase randômico conforme as equações 4.6 e 4.7 [FAB2004b]. A modelagem mostra que o efeito do *time jitter* se reflete na banda de sinal como um agregado de ruído, provocando uma degradação do SNR do sinal na saída do mixer. Na figura 4.6, é mostrado o efeito desta imperfeição.

Por sua vez, o modulador ΔΣ foi modelado empregando uma abordagem hierárquica de tempo discreto a partir de um modulador ΔΣ passa-baixas, conforme figura 4.9. A partir do protótipo discreto passa-baixas empregou-se a transformação passa-baixas para passa-banda clássica de filtros [VAZ2003]. A passagem do domínio tempo discreto para o contínuo é feita através da expressão 4.21, que estabelece a equivalência dos dois sistemas empregando invariância ao pulso. Com esse procedimento, obteve-se a função de conformação de ruído desejada, permitindo calcular a densidade espectral e potência total de ruído de quantização na banda do sinal, dadas respectivamente pelas equações 4.26 e 4.27. Empregando expansão em série de Taylor ao redor da freqüência central foi possível obter uma expressão fechada para a potência total do ruído de quantização na banda de interesse bem como o respectivo SNR do modulador, dados pelas expressões 4.29 e 4.30, respectivamente. As expressões que relacionam o desempenho do modulador quanto ao ruído de quantização têm como parâmetros a taxa de sobre-amostragem (OSR), a ordem do modulador (N), o número de bits do quantizador (B) e o fundo de escala do quantizador (Q_{FS}).

A combinação do modelo de desempenho do mixer e do modulador ΔΣ resulta na expressão 4.33 que sintetiza o desempenho global da FAC.

Foram desenvolvidos, além do modelo teórico, modelos de simulação tanto de tempo discreto como de tempo contínuo, conforme mencionado nos capítulos 4 e 5 e apêndices A e B. Esses modelos de simulação serviram para auxiliar na validação do modelo proposto bem como na etapa de projeto e depuração dos protótipos da FAC.

O capítulo 5 relata o desenvolvimento de um protótipo da FAC desde a exploração do espaço de projeto, definição da arquitetura do modulador, implementação e caracterização. Na realidade, foram desenvolvidos dois protótipos. O primeiro foi um protótipo de ordem 1 operando em freqüência baixa (freqüência central ao redor de 1 kHz), este foi utilizado como prova de conceito da FAC [FAB2003a, FAB2003b e FAB2004a]. O segundo protótipo construído tinha como desempenho alvo o processamento de sinais com largura de banda máxima de 60 kHz cuja freqüência central estaria entre DC e 20 MHz. Este último deveria fornecer uma resolução mínima equivalente a 8 bits (ENoB) correspondendo a um SNR de 50 dB. Empregando o modelo desenvolvido para FAC, a exploração do espaço de projeto foi realizada para prever o máximo *jitter* temporal admissível no mixer e a ordem, a freqüência de amostragem e o OSR do modulador ΔΣ. O resultado da exploração do espaço de projeto resultou nas seguintes especificações para a FAC:

- máximo *jitter* temporal no mixer de 16 ps_{RMS};
- ordem do modulador ΔΣ igual a 2 (dois ressonadores);
- quantizador de 1 bit (comparador);
- OSR mínimo de 32;
- freqüência central de processamento $\mathbf{f_P}$ de 1MHz e freqüência de amostragem igual a 4MHz.

Um protótipo da FAC com as características ou especificações acima foi construído, depurado e validado. Os resultados experimentais mostram uma boa conformidade com o modelo de predição teórica e de simulação.

A figura 5.10 mostra a densidade espectral de potência do *bit-stream* digital e sua conformidade com o modelo simulado quando as características de *jitter* temporal no

mixer e o ganho finito dos amplificadores operacionais do modulador ΔΣ são levados em consideração.

A caracterização da FAC em função da freqüência do sinal de entrada é mostrada na figura 5.15. Observa-se um desvio e uma queda de desempenho à medida que a freqüência do sinal de entrada aumenta. Isto se deve à existência de um jitter temporal medido de $240ps_{RMS}$ tanto nos sinais de controle do mixer como do modulador. Essa incerteza temporal tem origem na geração do sinal de relógio da placa com o FPGA agregada a FAC. O *jitter* temporal nas linhas de sinais de controle se mostrou um fator de forte impacto negativo no desempenho global da FAC. A limitação da máxima freqüência central do sinal de entrada em torno de 17 MHz neste protótipo resultou da máxima freqüência possível de ser gerada na placa contendo o FPGA e não da FAC que, por projeto, poderia chegar até aproximadamente 50 MHz.

Além da caracterização da FAC, algumas aplicações mapeadas sobre a arquitetura proposta são apresentadas também no capítulo 5. A primeira aplicação relatada é a utilização da FAC como um conversor analógico-digital multi-bandas parametrizáveis, cujos resultados estão sintetizado na tabela 5.2. Esta aplicação mostra que o usuário pode definir uma relação de compromisso entre resolução (SNR), potência e área (OSR) no tempo de desenvolvimento da aplicação. A interface tem um desempenho praticamente constante ao longo da freqüência central do sinal de entrada. São também apresentadas aplicações tais como multiplicação e adição de sinais empregando o domínio ΔΣ [FAB2004a].

As contribuições, conclusões e trabalhos futuros são apresentados no capítulo 6.

As principais contribuições deste trabalho foram:

- o desenvolvimento de uma arquitetura de interface fixa (FAC) e seu modelamento tanto teórico como de simulação;
- o modelo para exploração do espaço de projeto da célula analógica fixa (FAC);
- o desenvolvimento do modelo paramétrico da FAC;
- a caracterização da interface proposta;
- o desenvolvimento de aplicações exemplo.

A interface proposta tem a capacidade de processar sinais de DC até altas freqüências empregando o conceito de translação em freqüência, propiciando o processamento homogêneo de sinais. O uso domínio ΔΣ para representação dos sinais propicia a reconfiguração estrutural no domínio digital. O emprego de uma célula analógica praticamente fixa facilita o processo de migração tecnológica.

As principais limitações desta interface residem em dois campos, um relacionado com a resposta em freqüência e outro com o comportamento próximo de DC. O primeiro caso diz respeito principalmente à situação de processamento de sinais com conteúdo espectral indesejado nas freqüências imagens descritas pela equação 3.5. Havendo sinais ou ruído nestas posições espectrais os mesmos serão realocados para a freqüência de processamento causando interferência e degradação do SNR. Já no segundo caso, a existência de um nível de *offset* no bloco de entrada limitará o processamento de sinais de amplitude na ordem de grandeza do *offset*, ou mesmo reduzindo o range dinâmico. Isto ocorre devido ao ajuste do ganho ser realizado através

do amplificador de ganho variável após o mixer, portanto amplificando também este sinal espúrio.

Os trabalhos de pesquisa futuros relacionados ao tema desta tese serão principalmente os relacionados com:

- o aprimoramento do modelo da FAC;
- a avaliação de outras topologias para o mixer e para o modulador ΔΣ;
- o desenvolvimento de uma versão integrada da FAC;
- a investigação de arquiteturas de sintetizadores de freqüência compatíveis a arquitetura proposta;
- o estudo de adequar a FAC para testabilidade;
- o desenvolvimento de aplicações adicionais.

O apêndice A apresenta alguns detalhes referentes ao modelo teórico da interface e à influência do produto ganho-faixa finito dos amplificadores operacionais na resposta dos ressonadores do modulador ΔΣ. Já o apêndice B descreve sucintamente o modelo de simulação desenvolvido e os detalhes relacionados ao projeto e depuração do protótipo descrito no capítulo 5.