الجـــمــهـوريـــة الجـزائـريـة الديمـقـراطيـة الـشـعـبيـة

PEOPLE's DEMOCRATIC REPUBLIC OF ALGERIA

وزارة التـــعــليـــم العـــــالي والبـــحث العــــلــمـــي

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

جـــــــــــامعة سعيدة – د. الطاهر مولاي –

UNIVERSITY OF SAIDA - Dr. MOULAY TAHAR

Faculty of Mathematics, Computer Science and Telecommunications

A Dissertation Submitted in Partial Fulfilment of the Requirements for a master's Degree in Telecommunications

Specialization: Telecommunications Systems

Presented by: Ms. MIMOUNI Karima
Ms. FARHI Assia

Multiresolution analysis applied to SDR signals using GNU Radio

Defended on 13 June,2026 in front of the jury composed of:

Dr. TAMI ABDELKADER	MCA	President
Dr. OUARDI AISSA	MCB	Supervisor
Dr. OUESSAI ASSMAA	MCA	Examiner

2025 / 2026

Acknowledgments

First and foremost, we would like to thank God for giving us the health, determination, and patience to complete our master's degree and produce this modest work.

We would like to thank Mr. OUARDI AISSA for agreeing to support us and for his help and tireless availability.

We would like to express my deep gratitude to Mr. TAMI Abdelkader and Ms. OUESSAI Asmaa for agreeing to devote their time and expertise to evaluating this work. Their commitment and participation have been an honor for me.

We would also like to thank all the professors in the Telecommunications Department, particularly those in the Telecommunications Systems track, who have been of invaluable assistance to us.

Dedication

With eternal gratitude and love, I dedicate this work to my beloved mother, Fatiha , the pillar of my life, for her endless sacrifices, boundless love, and prayers that guided me through every step of this journey.

To my dear father, Mohamed, whose wisdom, constant encouragement, and unwavering support have been the foundation of my academic success.

To my wonderful brothers, Mimoune and Taha Amine , thank you for your unconditional love, continuous support, and for always being my

source of strength.

A special and heartfelt thanks to my wonderful partner, Assia , for her exceptional cooperation, dedication, and support throughout this project.

To all my respected teachers, my classmates, and everyone who, directly or indirectly, supported and encouraged me throughout my years of study.

Karima MIMOUNI

Dedication

I dedicate this work to my beloved parents, Moudjilali and Zaggai, whose unconditional love, endless support, sacrifices, and encouragement have been the foundation of my success. Their wisdom, patience, and confidence in me have inspired me throughout my academic journey.

To my dear husband, Hamada, thank you for your constant support, understanding, patience, and encouragement. Your presence has been a source of strength and motivation during every step of this journey.

To my beloved sisters and brothers, thank you for your love, support, and encouragement.

Special thanks to my partner, Karima, for her cooperation and support, and to my dear best friend, Fatima, for her sincere friendship and encouragement

To all of you, I express my deepest gratitude and dedicate this achievement with love and appreciation.

Assia Farhi

Abstract

This dissertation addresses the design, implementation, and optimization of a signal analyzer based on Multiresolution Analysis (MRA) via discrete wavelet transform, integrated within a Software-Defined Radio (SDR) architecture. Although the structural flexibility of SDR platforms provides unprecedented agility for digitizing RF signals, processing non-stationary streams highlights the inherent limitations of classical Fourier analysis regarding its rigid time-frequency plane tiling.

The primary objective of this research is to overcome the boundary distortion (edge effects) induced by time-domain block segmentation during digital convolution operations within Conjugate Quadrature Filter (CQF) banks. The integration and experimental validation of these concepts were carried out using the open-source development framework GNU Radio, utilizing both simulated deterministic signals and real physical spectrum captures from an RTL-SDR hardware interface.

Experimental results demonstrate that raw block-based processing generates critical discontinuities and massive amplitude drops at the data window boundaries. Evaluation of boundary regularization techniques shows that while the Zero-Padding method mitigates edge truncation artifacts, it introduces transient micro-dropouts that degrade reconstruction fidelity. Conversely, the Overlap technique successfully eliminates edge distortions and guarantees seamless temporal continuity, at the expense of computational overhead and increased buffering latency.

Finally, an optimized alternative based on a continuous, Stream-Based Multiresolution Analysis architecture is proposed. By deploying cascade filter structures directly onto the uninterrupted sample stream, this methodology natively removes block-boundary discontinuities while minimizing overall latency. This paradigm establishes a robust framework for real-time wavelet thresholding denoising algorithms and polyphase implementations within broadband RF data streams.

Résumé

Ce travail de fin d'études porte sur la conception, l'implémentation et l'optimisation d'un analyseur de signaux basé sur l'Analyse Multirésolution (AMR) par transformée en ondelettes discrète, intégré au sein d'une architecture de Radio Logicielle (SDR). Bien que la flexibilité des plateformes SDR offre une grande agilité pour la numérisation des signaux RF, le traitement de flux non stationnaires se heurte aux limites de l'analyse spectrale classique de Fourier et à la rigidité du partitionnement temps-fréquence.

L'objectif majeur de cette thèse est de résoudre le problème des effets de bord engendrés par la segmentation temporelle en blocs lors des opérations de convolution numérique au sein des bancs de filtres miroirs en quadrature conjuguée (CQF). L'intégration et la validation expérimentale de ces concepts ont été menées dans l'environnement de développement *open-source* GNU Radio, en exploitant à la fois des signaux déterministes simulés et des captures de signaux réels issues d'une interface matérielle RTL-SDR.

Les résultats obtenus démontrent que le traitement brut par blocs induit des ruptures de continuité et des distorsions d'amplitude critiques aux extrémités des fenêtres temporelles. L'évaluation des techniques de régularisation aux frontières montre que la méthode d'extension par *Zero-Padding* atténue les sauts d'amplitude mais engendre des micro-coupures résiduelles préjudiciables à la fidélité du signal. En revanche, la technique de recouvrement (*Overlap*) élimine efficacement les distorsions de bord et garantit une parfaite reconstruction, au prix d'une augmentation de la charge de calcul et de la latence.

Enfin, une architecture optimisée d'Analyse Multirésolution Continue (*Stream-Based*) est proposée. En appliquant directement les bancs de filtres sur le flux ininterrompu d'échantillons sans vectorisation préalable, cette approche native supprime définitivement l'apparition des effets de bord tout en minimisant la latence globale du système. Ce paradigme ouvre des perspectives solides pour l'intégration en temps réel d'algorithmes de débruitage par seuillage d'ondelettes et d'implémentations polyphasées au sein des futures normes de télécommunications.

ملخص البحث

تتناول المذكرة الجامعية هذه تصميم وتنفيذ وتحسين محلل إشارات يعتمد على التحليل متعدد عبر تحويل الموجات الصغيرة المنفصلة، مدمجًا ضمن بنية الراديو المعرَّف (MRA) الدرجات توفر قدرة غير مسبوقة على SDR ورغم أن المرونة الهيكلية لمنصات .(SDR) بالبرمجيات رقمنة إشارات التردد اللاسلكي، فإن معالجة التدفقات غير الثابتة تسلط الضوء على القيود المتأصلة في تحليل فورييه الكلاسيكي فيما يتعلق بتقسيمه الصارم لمستوى الزمان-التردد.

الهدف الأساسي من هذا البحث هو التغلب على تشوه الحدود (تأثيرات الحواف) الناتج عن تجزئة الكتل في المجال الزمني أثناء عمليات التحويل الرقمي داخل مجموعات مرشحات التربيع تم تنفيذ تكامل هذه المفاهيم والتحقق التجريبي منها باستخدام إطار العمل .(CQF) المرافق ، بالاستفادة من كل من الإشارات الحتمية المحاكاة والتقاطات GNU Radio المفتوح المصدر الطيف الفيزيائية الحقيقية من واجهة أجهزة .RTL-SDR

تُظهر النتائج التجريبية أن المعالجة الأولية القائمة على الكتل تولد انقطاعات حرجة وانخفاضات هائلة في السعة عند حدود نافذة البيانات. ويُظهر تقييم تقنيات تنظيم الحدود أنه في حين أن طريقة تخفف من آثار اقتطاع الحواف، فإنها تُحدث انقطاعات صغيرة عابرة تقلل من Zero-Padding تنجح في القضاء على تشوهات Overlap دقة إعادة البناء. وعلى العكس من ذلك، فإن تقنية الحواف وتضمن استمرارية زمنية سلسة، على حساب زيادة الحمل الحسابي

List of Figures

List of tables

Table of Contents

Abbreviation

SDR : Software-Defined Radio.

MRA : Multiresolution Analysis.

GNU Radio : plateforme de développement open-source.

CQF : Conjugate Quadrature Filters.

RTL-SDR : Realtek Software Defined Radio.

AMR : Analyse Multirésolution.

CWT : Continuous Wavelet Transform.

DWT : Discrete Wavelet Transform.

QMF : Quadrature Mirror Filters.

PR : Perfect Reconstruction.

RF : Radio Frequency.

IQ : In-phase / Quadrature.

I : In-phase.

Q : Quadrature.

ADC : Analog-to-Digital Converter.

DAC : Digital-to-Analog Converter.

DSP : Digital Signal Processing / Digital Signal Processor.

ASIC : Application-Specific Integrated Circuit.

FPGA : Field-Programmable Gate Array.

PC : Personal Computer.

IF : Intermediate Frequency.

LNA : Low-Noise Amplifier.

SNR : Signal-to-Noise Ratio.

TX : Transmission.

RX : Reception.

LO : Local Oscillator.

LPF : Low-Pass Filter.

MIMO : Multiple-Input Multiple-Output.

SoC : System-on-Chip.

RFIC : Radio Frequency Integrated Circuit.

RFSoC : Radio Frequency System-on-Chip.

ESD : Electrostatic Discharge.

LDO : Low-Dropout regulator.

HF : High Frequency.

STFT : Short-Time Fourier Transform.

WBFM : Wideband Frequency Modulation.

NCO : Numerically Controlled Oscillator.

GENERAL INTRODUCTION:

In today's telecommunications landscape, the agility of software and the power of digital signal processing have given rise to a new concept: Software-Defined Radio, or SDR. This technology was conceived in the early 1990s by Professor Joseph Mitola. The idea is to move modulation, demodulation, and filtering operations from the hardware domain, which until then had been limited by analog components, to the software domain. This shift offers unprecedented flexibility, allowing a single infrastructure to be remotely reconfigurable, multi-service, and interoperable with various communication standards.

However, the transition from the analog radio frequency (RF) signal received by the antenna to the digital processor imposes strict constraints on data conversion. The use of complex envelopes and the in-phase and quadrature (I/Q) structure has become a mathematical necessity for efficiently modeling and manipulating baseband signals as complex dummy signals. Furthermore, the transition to the digital domain using professional analog-to-digital converters (ADCs) requires adherence to the Nyquist-Shannon criterion to avoid aliasing, while simultaneously managing the critical trade-off between sampling rate and bit resolution. Modern SDR platforms, ranging from the cost-effective RTL-SDR receiver to highly integrated-on-chip architectures like the ADALM-PLUTO (***AD9361***) or cutting-edge solutions such as the Zynq UltraScale+ RFSoC (***Xilinx***), exemplify this ongoing effort to push the boundaries of digitization closer to the antenna.

Once the signal is digitized, extracting and analyzing information within non-stationary or noisy streams constitutes the core of current algorithmic challenges. While the classical Fourier transform remains the fundamental tool of frequency analysis, it proves structurally unsuitable for characterizing signals with evolving dynamics due to its inherent loss of temporal localization and the rigid limitations. It is to overcome this uniform tiling of the time-frequency plane that the Wavelet Transform was introduced, available in Continuous (CWT), Discrete (DWT), and Dyadic forms. By relying on dilation (scaling) and translation operations, wavelet analysis enables adaptive multi-resolution exploration, adjusting the size of the mathematical windows according to the frequency of the phenomena being studied.

Multiresolution analysis (MRA), theorized notably by Stéphane Mallat, formalizes this approach by proposing a hierarchical and iterative decomposition of the signal into approximation coefficients (low frequencies) and detail coefficients (high frequencies). This process is technically based on multirate filtering (decimation and interpolation) combined with conjugate quadrature mirror banks (CQF), theoretically guaranteeing perfect reconstruction of the signal without distortion. However, the practical application of this theory to finite digital signals encounters a major technological obstacle: ***edge effects***. During digital convolutions, truncation at the ends of data blocks generates critical disturbances and temporal continuity losses.

The aim of this dissertation is to explore the theoretical foundations of SDR technology and multiresolution analysis, and then to design, implement, and optimize algorithmic solutions capable of overcoming these edge distortions in real time. To achieve this, this work combines rigorous mathematical modeling and experimental validation using the open-source GNU Radio development environment and real-time spectrum capture (RTL-SDR) hardware.

Chapter 1 SDR Technology

1.1. Introduction to Software-Defined Radio

Software-defined radio (SDR) was proposed by Professor Joseph Mitola, who published the first article on the subject in 1992 [1]. This article puts forward a proposal for a new low-cost, low-power radio technology, enabling the realization of radio base station terminals and infrastructures capable of supporting, independently of the hardware, multi-service, flexible, multi-band, remotely reconfigurable and software-reprogrammable operation [2][3].

A software-defined radio (SDR) is a radio communication system that performs modulation and demodulation of the radio signal in software [1]. However, the underlying philosophy of the SDR concept is that the software should operate as close as possible to the antenna and run on a general-purpose computer. Figure 1.1 represents the block diagram of an ideal SDR. An SDR is a radio transceiver consisting of two parts: the software and the hardware. In reception, the RF front-end converts the received signal from its carrier frequency to an intermediate frequency (IF) or baseband. In transmission, the RF front-end converts the baseband signal to an intermediate frequency (IF) and then to the desired carrier frequency. In both cases, the analog-to-digital converter (ADC) converts the signal to the modulation bandwidth. The following processing steps (filtering, decimation, IQ demodulation, decoding, interpolation, IQ modulation, …) can then be performed in software. These processing steps are performed using a microprocessor dedicated to signal processing (DSP, digital signal processor), a component dedicated to signal processing (ASIC: Application-Specific Integrated Circuit), a programmable electronic component (FPGA, Field-Programmable Gate Array), or directly on the processor of a traditional PC [5].

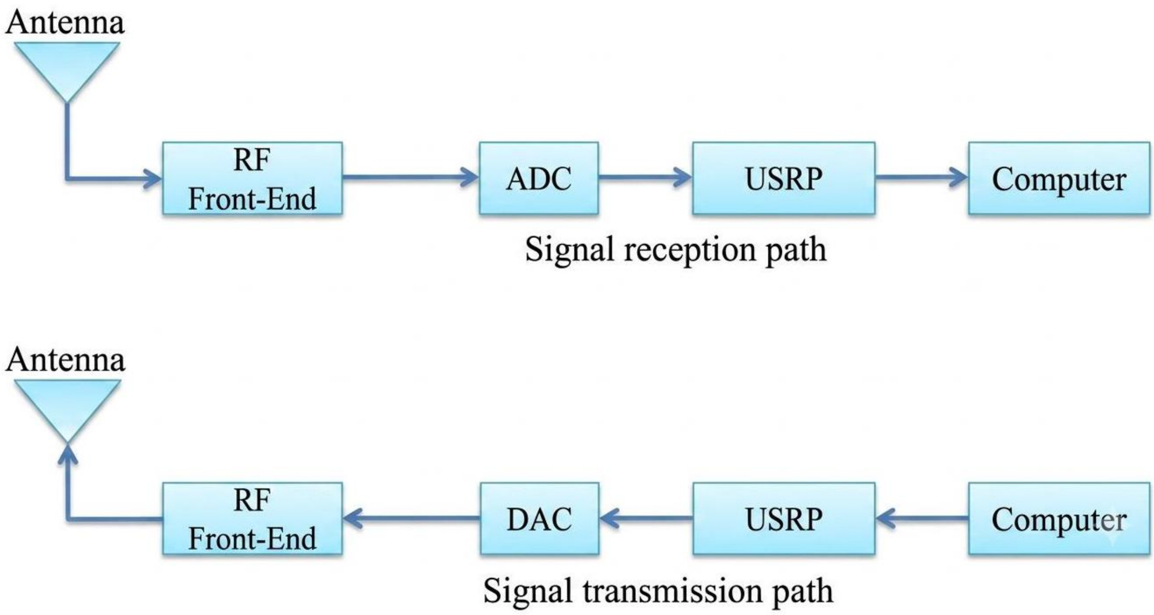


Figure 1.1: Diagram of an ideal SDR communication system

1.2. The complex envelope and IQ structure

1.2.1. I/Q Modulation:

The digital modulation operation can theoretically be subdivided into two steps: the construction of the baseband components of the modulated signal, followed by a transposition operation around the desired carrier frequency, called "I/Q Modulation" [6].

The modulated signal x(t) being a real bandpass signal of width B around a carrier frequency, it can always be written as a narrowband signal (B$\boldsymbol{f_0}$.<$\boldsymbol{f_0}$) :

$$\boldsymbol{x(t) = x_I(t) \cos(2\pi f_0 t) - x_Q(t) \sin(2\pi f_0 t)} \qquad \textbf{(1.1)}$$

Or$\boldsymbol{x_I(t)}$and are two real baseband signals (low-pass type, one-sided band [0,b/2]) called respectively in-phase (I, "In phase") and quadrature (Q, "Quadrature") channels because they modulate the carrier "cos (2$\boldsymbol{x_Q(t)B\pi f_0 t)}$" and the quadrature carrier" .$\boldsymbol{sin(2\pi f_0 t)}$

The term "I/Q modulation" refers to the conversion of the components $\{\boldsymbol{x_I}(\mathbf{t}), \boldsymbol{x_Q}(\boldsymbol{t})\}$ to a bandpass signal.$\boldsymbol{x_I}(\mathbf{t}), \boldsymbol{x_Q}(\boldsymbol{t}), \boldsymbol{x}(\boldsymbol{t})$.This operation corresponds to a universal "frequency transposition," which applies to both analog and digital modulations. In the case of digital modulation, the analog signal$\boldsymbol{x}(\boldsymbol{t})$It carries digital information, which is distinct across (t) on the one hand, and on the other. It will be convenient to represent this dual dimension using a complex low-pass signal [6]:$\boldsymbol{x_I}\boldsymbol{x_Q}(\boldsymbol{t})$

$$\tilde{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{x}_{\mathrm{I}}(\boldsymbol{t}) + \boldsymbol{j}.\boldsymbol{x_Q}(\boldsymbol{t}) \quad \textbf{(1.2)}$$

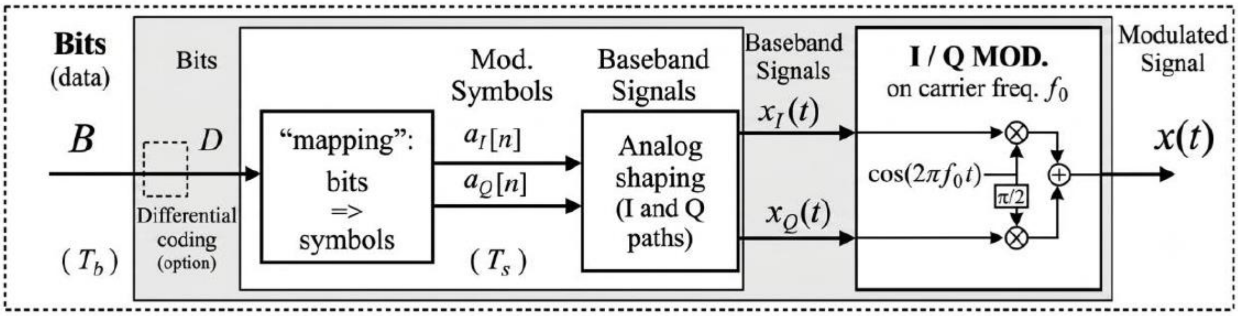


Figure 1.2: Block diagram of a digital carrier frequency modulation [6]

1.2.2. Representation of the I/Q modulated signal using the complex envelope

The study of modulation techniques is facilitated by the use of the complex envelope, which is a low-pass (baseband) signal. The actual modulated signal x(t), emitted in a band around the frequency, can be equivalently expressed by:$\tilde{\boldsymbol{x}}(\boldsymbol{t})\boldsymbol{f_0}$,

$$\boldsymbol{x}(\boldsymbol{t}) = \boldsymbol{Re}\{\tilde{\boldsymbol{x}}(\boldsymbol{t}).\boldsymbol{e}^{\boldsymbol{j2\pi f_0 t}}\} \quad \textbf{(1.3)}$$

With the complex envelope defined in Cartesian coordinates by:

$$\tilde{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{x}_{\mathrm{I}}(\boldsymbol{t}) + \boldsymbol{j}.\boldsymbol{x_Q}(\boldsymbol{t}) \quad \textbf{(1.4)}$$

Or :

- $\boldsymbol{x}_{\mathrm{I}}(\boldsymbol{t})$:is the in-phase component.

- $x_Q(t)$: is the quadrature component.

This representation combines the two real baseband components into a single complex fictitious signal, which greatly simplifies system calculations and modeling.radio [6].

1.3. Mixer and Analog-to-Digital Converter (ADC)

This section analyzes the transition from the radio frequency (RF) signal to the digital domain. This interface is crucial because it determines the accuracy and bandwidth usable by the SDR system.

1.3.1 Nature and properties of the information processed

Before any conversion operation, it is essential to characterize the incident signal:

- The analog dimension: As indicated by the fundamentals of signal processing, the physical world is analog in nature. The signal captured at the antenna is a continuous electromagnetic wave [4] [5].
- The signal as a voltage: More specifically, the information manifests as an electrical voltage present across the antenna terminals. This signal is continuous in time and amplitude, permanently defined for each instant t.$v_a(t)$
- Arguments for digitization: The shift to digital is motivated by the need to ensure excellent reproducibility of processing, to facilitate massive storage and to allow the implementation of complex mathematical functions at a reduced cost [5].

1.3.2. The mixer and frequency transposition:

In a conventional receiving architecture, the captured RF signal often has a frequency that is too high to be processed directly.

- Multiplication mechanism: The mixer performs a time multiplication between the RF signal and a sinusoidal signal from a local oscillator (LO) [5].
- Intermediate Frequency (IF) Transposition: This operation shifts the signal spectrum to a lower frequency (IF), which helps to reduce the speed constraints on the ADC conversion stage.

- SDR evolution: The goal of software-defined radio is to minimize these hardware blocks to perform mixing and filtering directly via digital algorithms [1] [7].

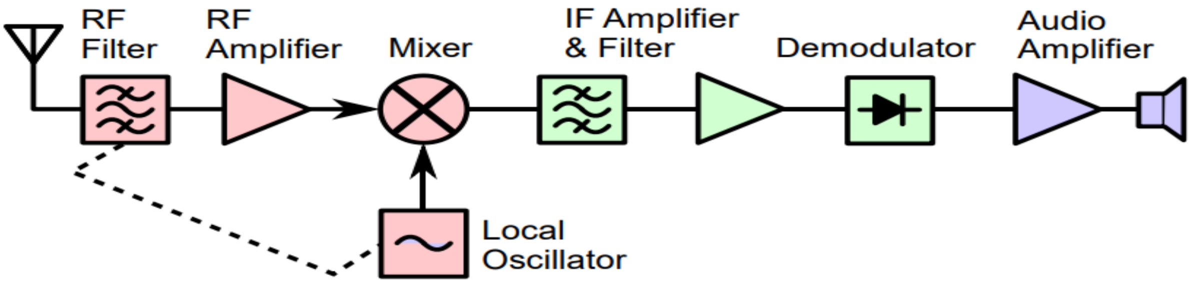


Figure 1.3: Functional architecture of a conventional radio frequency receiver[5]

1.3.3. In-depth analysis of the conversion process (ADC):

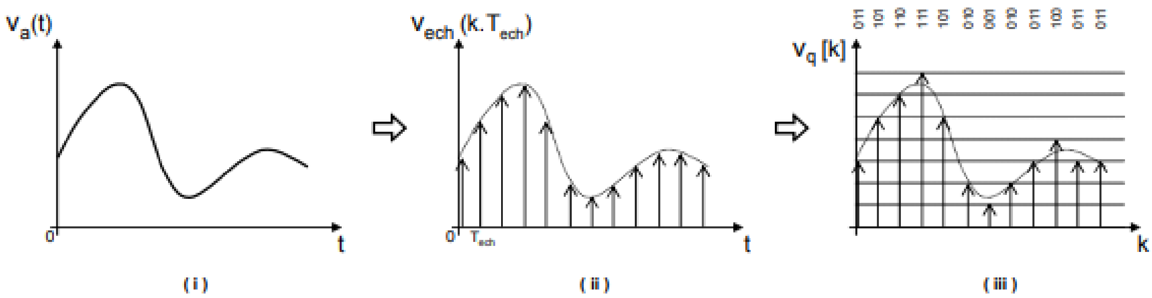


Figure 1.4: Representation of the digitization steps of a signal: (i) analog, (ii) sampled and (iii) quantized [7]

The analog-to-digital converter ensures the discretization of the signal x(t) following three rigorous steps [7]:

1.3.3.1. Time Sampling:

It consists of taking values from the signal at regular intervals. Mathematically, this amounts to multiplying x(t) by a Dirac comb (t)[8]:$\boldsymbol{T\delta_{T_{ech}}}$

$$x_{ech}(kT_{ech}) = x(t).\sum_{k=-\infty}^{+\infty} \delta(t - k.T_{ech}) \quad (1.5)$$

- Nyquist-Shannon criterion: To avoid aliasing, the sampling frequency must be greater than twice the maximum frequency of the signal ($f_s f_s > 2 . f_{max}$)If this condition is not met, the high-frequency components overlap with the low frequencies, making the information unrecoverable.

1.3.3.2. Quantification and Resolution:

This step transforms the continuous amplitude into discrete levels [7]. The number of levels is dictated by the resolution N (in bits) of the converter (levels). The quantization step q represents the minimum voltage resolution:$\mathbf{2^N}$

$$q = \frac{v_{max} - v_{min}}{2^N} [7] \quad (1.6)$$

- Impact on the signal-to-noise ratio (SNR):

The higher the N resolution, the lower the quantification error, which improves the dynamics of the SDR receptor [8].

1.3.3.3. Coding:

Each quantified sample is translated into a binary word. In SDR systems, this data is structured into I (In phase) and Q (Quadrature) components in order to preserve the integrity of phase and amplitude information before software processing [5].

1.4. Hardware Architecture of SDR Systems

Hardware architecture is the differentiating element of an SDR system. It ensures the transition between the analog radio spectrum and digital processing algorithms. The major challenge is to push the limits of digitization as close as possible to the antenna while managing dynamic range and noise constraints [1].

1.4.1. From ideal architecture to actual implementation

Theoretically, an ideal SDR would rely on direct digitization at the antenna output. This would require an analog-to-digital converter (ADC) capable of sampling frequencies of several gigahertz with a resolution of n bits [7].

In practice, the following architectures are used to adapt the signal:

- ***SuperheterodyneArchitecture***It uses a mixing stage to bring the signal back to an intermediate frequency (IF). It is the most efficient in terms of selectivity, but it remains complex to miniaturize [5].
- ***Zero-IF architecture***: The RF signal is directly transposed into baseband. This is the preferred topology in modern receivers (like the AD9361) because it allows a drastic reduction in the number of analog components (filters, mixers) and facilitates integration on an RFIC chip [5] [7].

1.4.2. Comparative study of reference hardware platforms

Analysis of current SDR platforms reveals a hierarchy based on resolution, bandwidth, and onboard computing power.

1.4.2.1. HackRF One: Broadband Spectral Agility

The HackRF One is a widely used SDR platform for radio spectrum exploration. Its hardware architecture, illustrated by the block diagram in Figure 1.5, is characterized by extreme flexibility but presents specific technical compromises [9]:

- Conversion architecture (Mixer-based): Unlike simple direct conversion architectures, the HackRF uses a structure based on an RFFC5072 RF mixer. This allows signals from the 1 MHz to 6 GHz range to be transposed to an intermediate frequency (IF) processed by the receiver [8].
- RF chain (MAX2837): The signal is then handled by the MAX2837 circuit. This wideband transceiver incorporates programmable low-pass filters and variable gain amplifiers (LNA/VGA) to optimize reception before digitization [9].
- MAX5864: is an RF transceiver integrated circuit designed for broadband wireless communication applications, particularly SDR systems. It integrates both high-speed

analog-to-digital converters (ADC) and digital-to-analog converters (DAC), enabling direct conversion between RF signals and digital data [9].

- Processing and Control (LPC4320): The board is controlled by a dual-core NXP LPC4320 microcontroller (Cortex-M4 and Cortex-M0). It manages the USB 2.0 interface for the transfer of IQ samples and ensures synchronization with the Si5351C clock generator [9].

Technical limitations:

- Resolution: The system uses an 8-bit resolution, which limits its dynamic range compared to other devices.
- Half-Duplex Mode: The diagram clearly shows that the transmit (TX) and receive (RX) paths share the same resources, thus preventing simultaneous bidirectional communication.

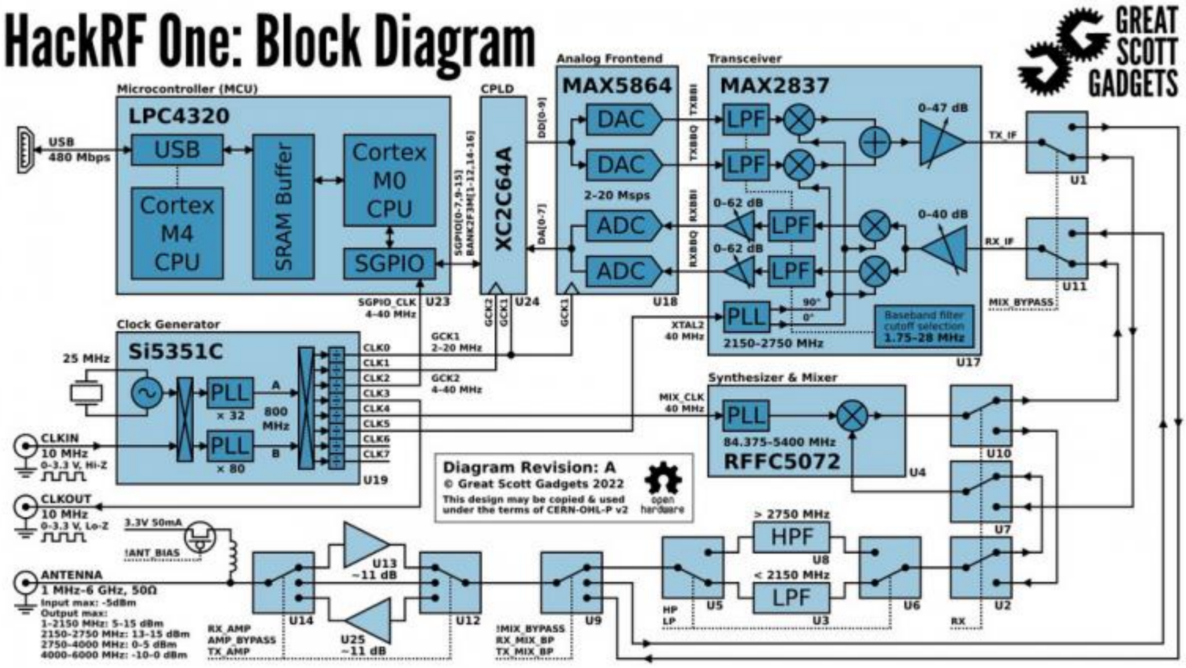


Figure 1.5: Functional block diagram of the HackRF One platform [9]

1.4.2.2. BladeRF 2.0 micro: FPGA Performance and Parallelism

The bladeRF 2.0 micro is distinguished by a hardware architecture optimized for wideband signal processing. As shown in the block diagram in Figure 1.6, the system is structured around three main modules [10]:

- Digital processing (Intel Cyclone V): Integrating this FPGA allows complex DSP functions, such as FIR filters and decimation stages, to be implemented directly in hardware. [10] This reduces the computational load on the host computer.
- RF section (Analog Devices AD9361): This RFIC transceiver provides signal conversion in Full-Duplex MIMO 2x2 mode. It offers 12-bit resolution and supports an instantaneous bandwidth of up to 56 MHz, making it suitable for LTE and 5G standards [10].
- Data interface (Cypress FX3): The USB 3.0 controller ensures the transfer of IQ sample streams between the board and the host, thus ensuring sufficient throughput for broadband applications [10].

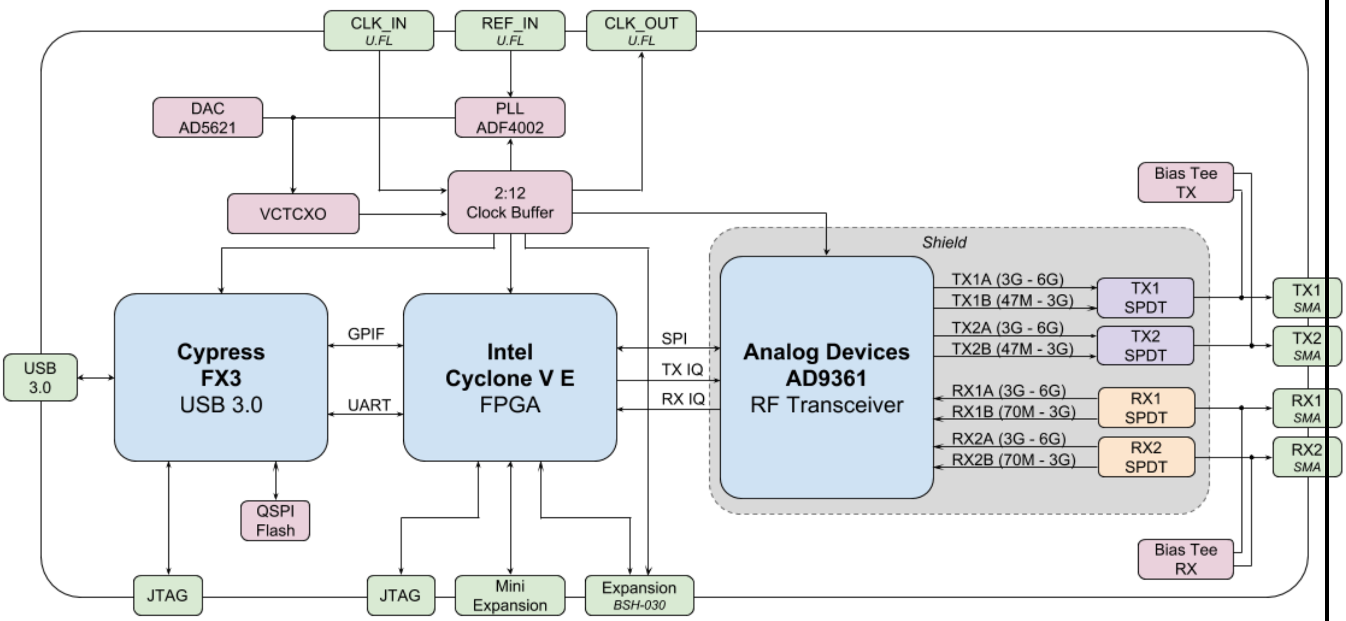


Figure 1.6: Hardware block diagram of the bladeRF 2.0 micro board [10]

1.4.2.3. ADALM-PLUTO: The System-on-Chip (SoC) approach:

The PlutoSDR is based on a highly integrated architecture thanks to the AD9361 circuit[7]. The functional diagram in Figure 1.7 allows for analysis of the internal complexity of the processing chain[11]:

- RF Front-end Architecture: Unlike superheterodyne architectures, the AD9361 uses a direct conversion (Zero-IF) topology. The RF signal passes through a low-noise amplifier (LNA) stage before being directly demodulated into baseband (I and Q) by quadrature mixers [11].
- Conversion chain (ADC/DAC): The circuit integrates 12-bit Sigma-Delta converters. This resolution is crucial to guarantee a high dynamic range and better noise immunity compared to 8-bit systems (Example: RTL-SDR) [7].
- Internal Digital Filtering: Before reaching the processor, the signal undergoes intensive digital processing inside the chip (decimation stages, programmable FIR filters, etc.). This allows for precise adjustment of the channel bandwidth from 200 kHz to 56 MHz [11].
- The Processing Core (Zynq-7000): The transceiver is interfaced with a Xilinx Zynq-7000 SoC, combining an FPGA for high-speed signal processing and an ARMCortex-A9 processor for managing the software stack and communication protocols [11].

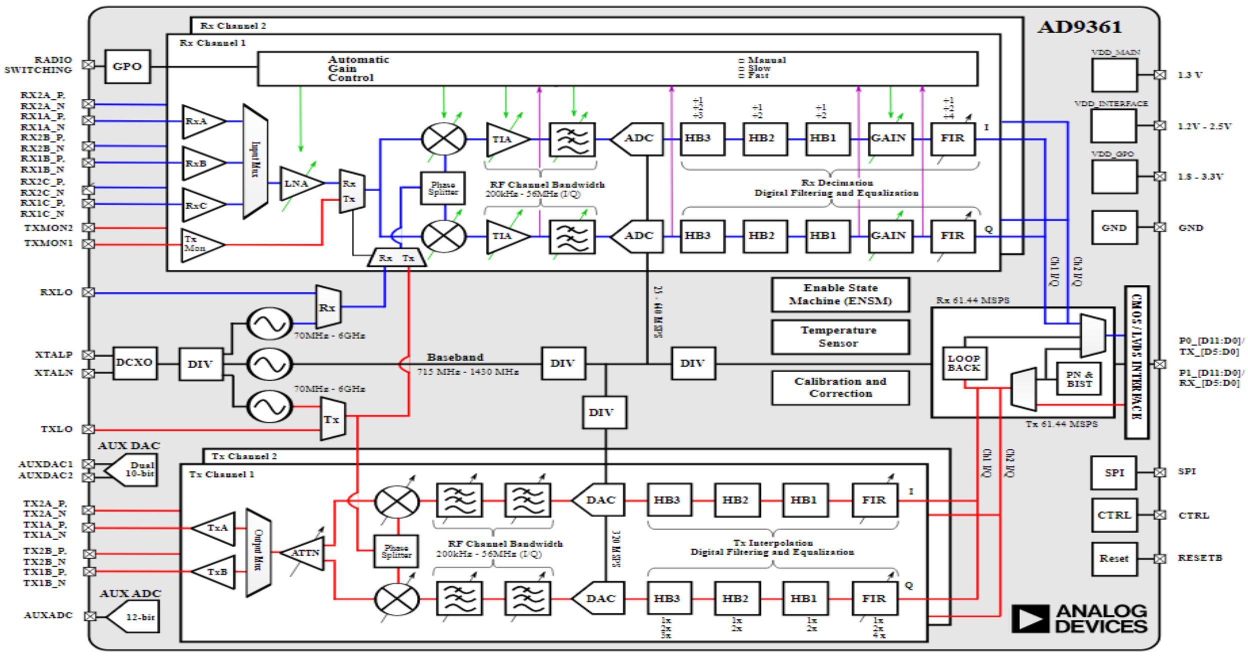


Figure 1.7: Detailed internal architecture of the AD9361 circuit

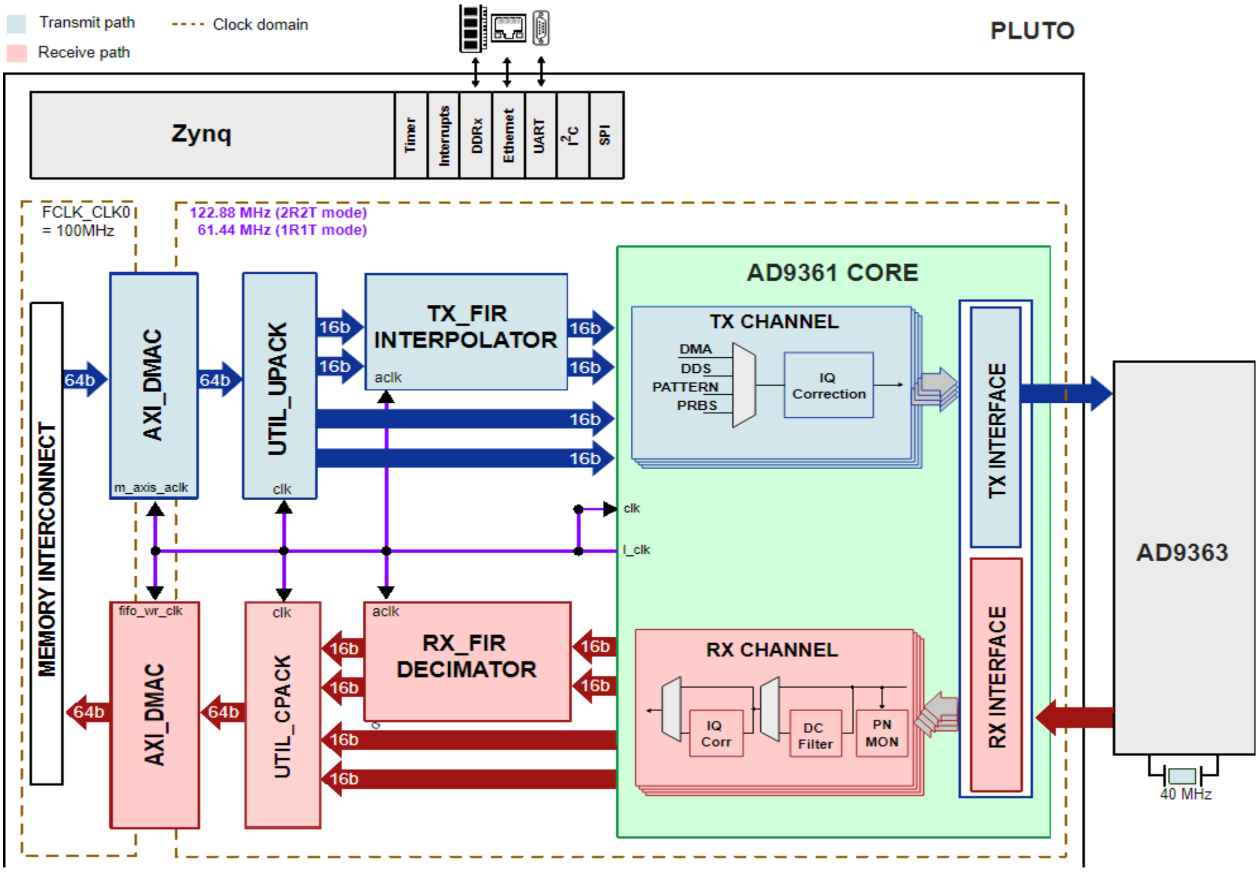


Figure 1.8: Functional architecture and data flow of the ADALM-PLUTO system [11]

1.4.2.4. RTL-SDR: Low-Cost Reception:

The RTL-SDR is based on the RTL2832 chipset (Figure 1.9). This component performs the digital signal processing after the R820 tuner stage[12]:

- Digital Demodulation: The digitized signal passes through digital mixers and low-pass filters (LPF) for the extraction of the I and Q components (Circuit R820) [12].
- Data formatting: Samples are coded on 8 bits (I: 8bit, Q: 8bit) and then routed to the host via the USB controller [12].
- Performance: This architecture limits the dynamic range to approximately 48dB, restricting the use of this device to the reception of strong signals or to SDR initiation [12].

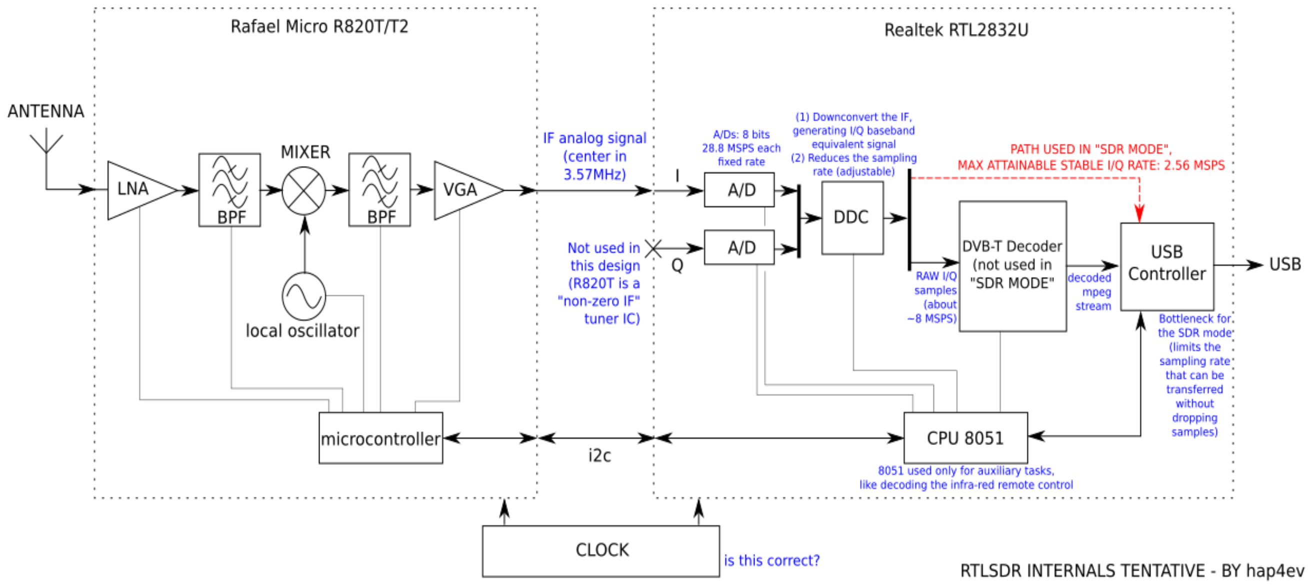


Figure 1.9: Detailed functional diagram of the RTL-SDR receiver (R820 tuner and RTL2832U interface) [12]

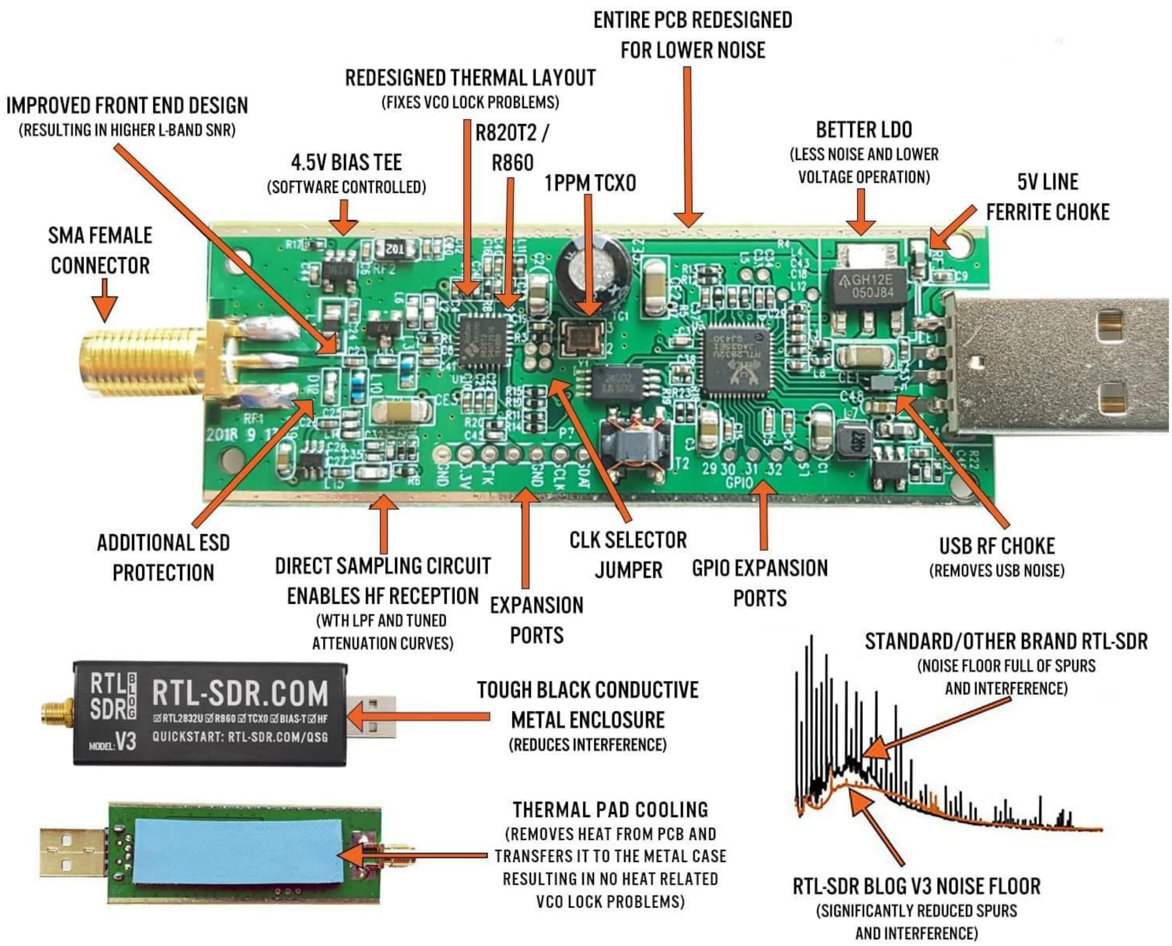


Figure 1.10: Hardware architecture of the RTL-SDR v3 Dongle [12]

The hardware implementation of the RTL-SDR (v3) dongle shown in Figure 1.10 reveals an architecture optimized for wideband reception. Unlike standard versions, this model incorporates a temperature-compensated oscillator to minimize frequency drift. The circuit is distinguished by the use of a robust SMA antenna interface, an electrostatic discharge (ESD) protection stage, and a low-noise voltage regulator (LDO). These components not only improve the signal-to-noise ratio (SNR) but also facilitate the activation of Direct Sampling mode for HF band reception via the GPIO expansion pins.

1.4.2.5 Zynq UltraScale+ RFSoC: SDR Dream

The RFSoC (Radio Frequency System-on-Chip) architecture is distinguished by the direct integration of the conversion chains on the same chip as the programmable logic. As illustrated in the block diagram in Figure 1.11, the system relies on direct digitization [13]:

- Direct RF sampling: The diagram shows that the RF data converters (ADC and DAC) are integrated into the silicon. This configuration eliminates the need for external analog mixers and allows for the processing of signals at sampling rates up to 5 Gsps[13].
- Digital Processing and FPGA: The data stream from the ADCs is processed by hardware blocks implemented in programmable logic (Matlab Generated HDL), including FFT, filtering and equalization (Equalizer Demod).
- Control unit (ARM Processor): The diagram highlights an ARM Processor dedicated to system management and register control (Register File), ensuring coordination between the software part and real-time signal processing [13].
- Antenna-to-Digital Architecture: The direct link between the RF Array and the integrated converters reduces latency and power consumption by eliminating complex serial interfaces [13].
- Resolution: The system operates with a 12-bit resolution (or 14 bits depending on the generation), ensuring high dynamic range for advanced telecommunications applications. The RFSoC architecture optimizes energy efficiency by reducing power consumption by 50% to 75%. This power saving is primarily due to the elimination of complex serial interfaces required by traditional discrete component solutions [13].

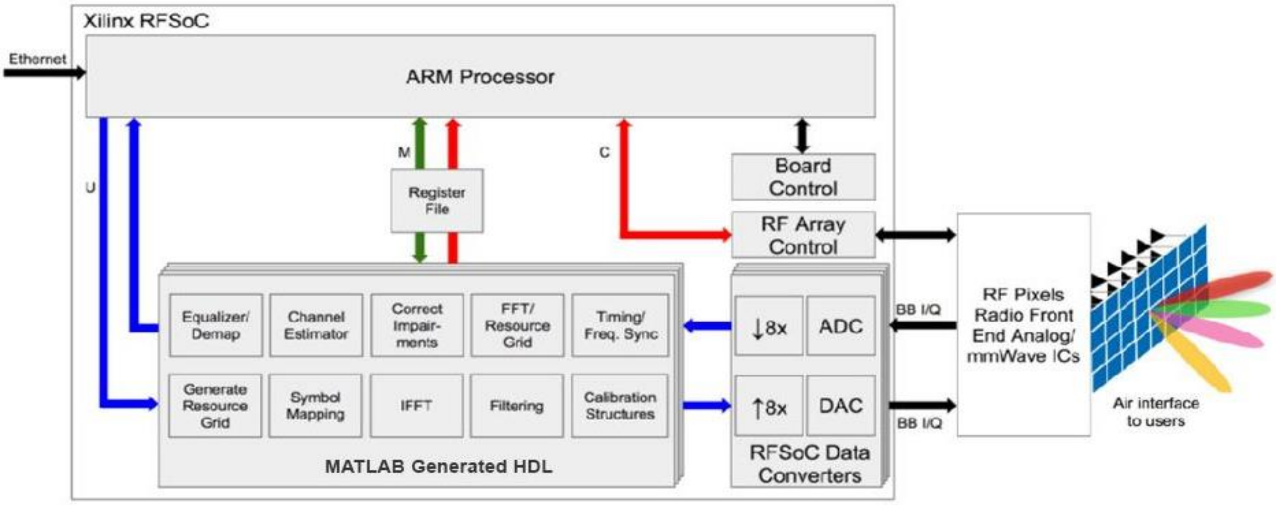


Figure 1.11: Architecture of an SDR system based on the Zynq UltraScale+ RFSoC

This platform is distinguished by:

- RF Data Converters: Integration of multiple ADC/DAC channels supporting direct sampling up to 4-6.4 Gsps, enabling direct capture in RF band [13] (up to 4 GHz bandwidth).
- Heterogeneous System (PS): A combination of a Quad-core ARM Cortex-A53 processor (1.5GHz) for the upper layers and a Dual-core Cortex-R5 for critical real-time tasks [13].
- Programmable Logic (PL): An UltraScale+ class FPGA dedicated to intensive processing (FEC, complex modulation).

Digital processing chain: The inclusion of DUC/DDC blocks with complex mixers and 48-bit digital oscillators (NCOs) for maximum frequency agility.

1.5. Professional ADCs (Sampling Frequency > 100 MHz)

The integration of analog-to-digital converters (ADCs) operating at rates exceeding 100 MSPS represents a major technological challenge for wideband SDR receivers. At these frequencies, the Pipeline architecture is consistently preferred. This structure allows for maintaining high resolution (up to 16 bits) by distributing quantization across multiple stages, thus ensuring optimal accuracy even when processing massive data streams [14].

The true performance of a professional ADC is not limited to its sampling rate, but also depends on its ability to preserve signal integrity in a crowded spectrum. For example, the AD9446 stands out with an SNR of 79.7 dBFS (at 10 MHz), a critical value for detecting low-amplitude signals buried in thermal noise [15].

Model	**Resolution**	**Maximum cadence ()f_s**	**SNR (dBFS)**	**Power (W)**	**Output standard**
AD9446	16 bits	100	79.7	2.6 W	LVDS
ADS5263	16 bits	100	85	1.4 W	CMOS or LVDS
AD9689	14 bits	2600	>80	1.55 W/channel	JESD204B

Table 1.1: Comparative study of the technical performance of industrial ADC converters for SDR systems [14] [15]

1.6. Resolution of the ADC and DAC

Resolution is the fundamental parameter that defines the accuracy of quantization and determines the overall dynamic range of the SDR system. It represents the smallest variation in analog voltage that can be discriminated by the ADC or generated by the DAC. High resolution is essential to lower the noise floor, thus enabling the extraction of low-amplitude signals in the presence of strong interference [7].

The conversion process is governed by the quantization step size (or resolution R), which defines the voltage difference between two successive digital codes. The response of an ADC follows a "stepped" curve. For a 3-bit converter with a full-scale range of 1 V, this results in a step size of 0.125 V. The maximum quantization error is defined by [5]:$\boldsymbol{q}V_{FH}E_{Q(MAX)}$

$$E_{Q(MAX)} = \frac{V_{FH}}{2^n \times 2} \quad \textbf{(1.7)}$$

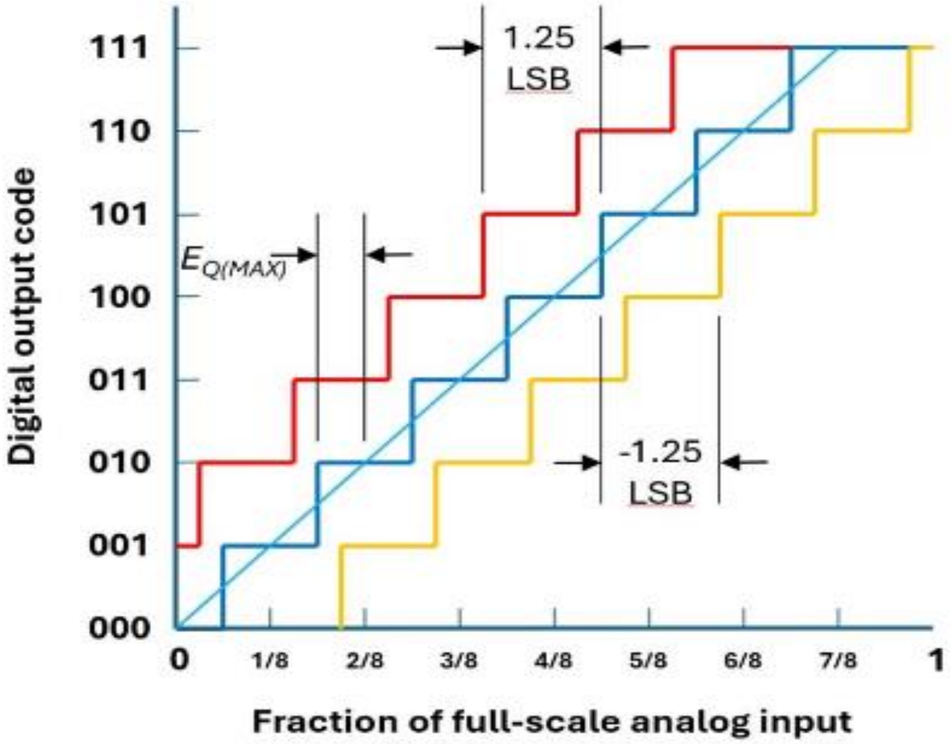


Figure 1.12: Transfer characteristic of an ADC and quantization error

In this specific case, the value is 62.5 mV (or 1/2 LSB). The figure also highlights positive (red) and negative (yellow) offset drifts relative to the ideal response (blue). Conversely, platforms like Xilinx's RFSoC use 16 bits, reducing this step size to a few microvolts.$E_{Q(MAX)}$

The choice of an SDR architecture depends on the trade-off between resolution and sampling rate. Performance varies drastically depending on the hardware:$\boldsymbol{f_s}$

- HackRF One: Limited to 8 bits, which restricts its use to simple monitoring [9].
- BladeRF 2.0: Offers 12-bit processing, enabling the handling of standards such as LTE [10].

- Xilinx RFSoC: Represents the state of the art with 16 bits. It enables Direct RF Sampling [13].

The transition from the analog to the digital domain does not depend solely on amplitude resolution, but relies primarily on the accuracy of the time capture. To guarantee faithful reconstruction without loss of information, the Nyquist criterion requires that the sampling frequency be strictly greater than twice the maximum frequency of the signal > 2 [8].$f_s \boldsymbol{f_s} \boldsymbol{f_{max}}$

The architecture of the Zynq UltraScale+ RFSoC is distinguished by its ability to integrate Direct RF converters. By achieving speeds of several GS/s, it eliminates the need for analog mixing stages while largely meeting Nyquist requirements for ultra-wide bandwidths [13].

Chapter 2: Wavelet Transformation

2.1 Limitations of Fourier analysis:

Classical Fourier analysis, although it forms the basis of frequency signal processing, has major theoretical and practical limitations when the signal under study deviates from the framework of stationarity. This section outlines the theoretical limitations of the FT, particularly with regard to the Heisenberg-Gabor uncertainty principle, as highlighted by Mallat [16].

2.1.1 Overall analysis and loss of temporal localization:

The Fourier Transform (FT) relies on an integral decomposition of the signal x(t) onto a basis of sinusoidal functions with infinite time support. Mathematically, for a signal with finite energy, it is defined by the following expression:

$$\boldsymbol{X(f) = \int_{-\infty}^{+\infty} x(t) e^{-i2\pi f t} dt} \qquad \textbf{\textit{(2.1)}}$$

Analysis of this expression shows that the basis of sinusoidal functions has infinite time support. Therefore, any local modification of the signal affects its entire spectrum, which makes the FT unsuitable for characterizing signals with evolving dynamics (non-stationary) [17].

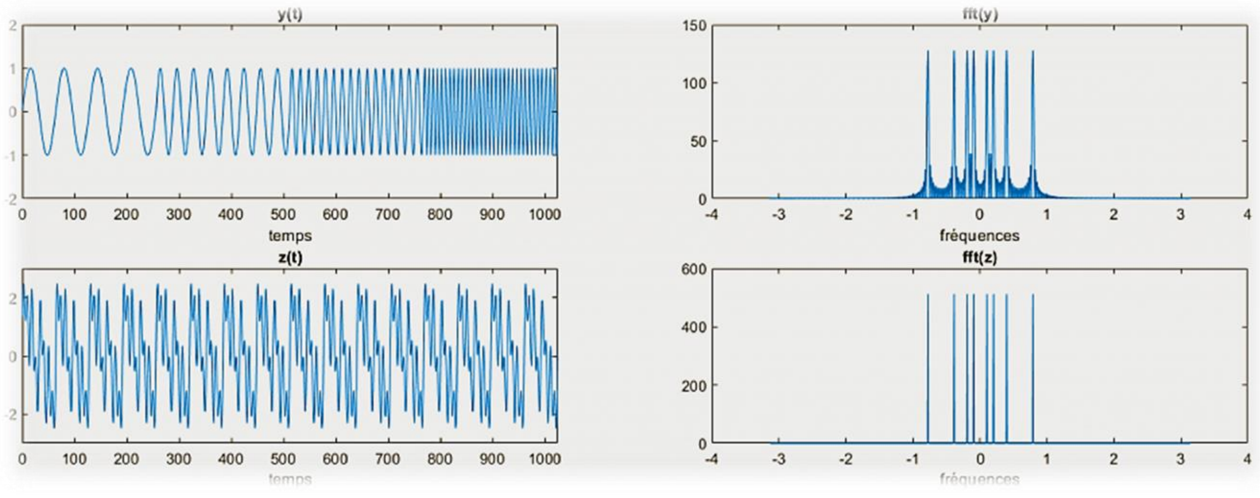


Figure 2.1: Comparison of two signals with different temporal structures but identical spectra [18]

This ambiguity is highlighted by the analysis of the results in Figure 2.1. By comparing the signal y(t) (sequential frequency components) and the signal z(t) (simultaneous frequencies), we observe that their respective spectra |Y(f)| and |Z(f)| are practically identical. This observation demonstrates that the classical Fourier Transform masks the chronological order of events, which justifies its unsuitability for processing signals whose statistical properties vary over time (non-stationary signals) [16].

2.2 Continuous Wavelet Transformation (CWT):

To overcome the limitations of STFT, and particularly its rigid tiling of the time-frequency plane, Continuous Wavelet Transform (CWT) was developed. Unlike STFT, CWT relies on multi-resolution analysis, allowing the size of the analysis windows to be dynamically adapted to variations in the signal [16].

2.2.1 Definition and mathematical principle:

The CWT (Constant Wavelet Time) of a signal is defined as the correlation between the signal and a family of wavelets. These wavelets are generated from a single parent wavelet (t) by the operations of dilation (scale) and translation (time). Mathematically, it is expressed by the following integral:$x(t\Psi_{a,b}(t).\Psi$

$$\boldsymbol{CWT_x(a,b) = \frac{1}{\sqrt{|a|}}\int_{-\infty}^{+\infty} x(t)\Psi^{*}\,(\frac{t-b}{a})\,dt} \qquad \textbf{\textit{(2.3)}}$$

The analysis is based on three fundamental parameters:

- The mother wavelet (t): This is the basic pattern that must absolutely satisfy the admissibility condition (zero mean). This property ensures the existence of an inverse transformation [16], [17]Ψ.
- The b shift (Translation): It defines the time position of the wavelet along the signal, thus ensuring precise localization in time [19].
- Scaling: This parameter replaces the notion of frequency. A small value of 'a' compresses the wavelet (high frequencies), while a large value expands it (low frequencies) [21], [22].
- The factor: It ensures the conservation of signal energy across all analysis scales.$\boldsymbol{\frac{1}{\sqrt{|a|}}}$

2.2.2 Scaling and Translation Mechanism:

Continuous wavelet transform analysis relies on the dynamic manipulation of a single basis function, called the mother wavelet. This process of extracting frequency characteristics is based on two fundamental operations, illustrated graphically in Figure 2.3:

- Dilation (a> 1): When the scale factor increases (as shown in examples (t/2) and (t/4)), the wavelet undergoes time spreading. This configuration acts as a low-pass filter, allowing the capture of slow components and overall trends of the signal [21].$\Psi\Psi$
- Compression (a< 1): Conversely, a reduction in the scale factor leads to a contraction of the wavelet. This temporal compression offers a possibility of analyzing high frequencies as well as short transient phenomena [21].
- Translation (b): This parameter allows the wavelet to be translated along the time axis. This operation guarantees an exhaustive exploration of the signal, allowing each frequency event to be precisely located in time [22].

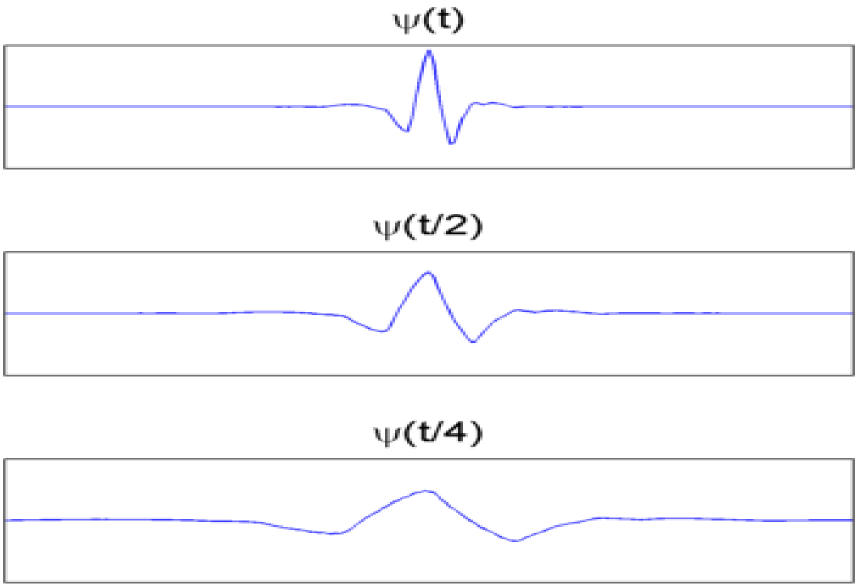


Figure 2.2: Influence of the scale factor () on the morphology of the mother waveletaΨ(t) [23]

2.3 Types of parent wavelets:

The choice of the analysis functionΨ(t)is crucial, as the quality of the decomposition depends on the morphological correlation between the signal being studied and the basis wavelet. In practice, there is no "universal" wavelet. Each family possesses specific characteristics (symmetry, number of zero moments, support) that make it optimal for a given type of signal.

Two main categories are distinguished: orthogonal families (used in DWT) and continuous families (CWT) [16], [24].

2.3.1 Orthogonal Families (Discrete Analysis DWT):

These wavelets are characterized by a compact support and allow decomposition without redundancy. They are fundamental for denoising and compressing signals [25].

- **Haar wavelet:**

Identified as early as 1909, the Haar wavelet is the oldest and simplest orthogonal basis. It is mathematically defined by a piecewise constant step function:

$$\Psi(t) = \begin{cases} \mathbf{1} & si\ \mathbf{0} \leq t < \frac{1}{2} \\ -\mathbf{1} & si\ \frac{1}{2} \leq t < 1 \\ \mathbf{0} & sinon \end{cases} \quad (2.4)$$

Although its discontinuous structure may limit the analysis of very smooth signals (because it is not differentiable), it offers unique advantages:

- Minimal support: It allows for perfect temporal localization and extremely fast algorithmic calculation.
- Break detection: It excels in identifying sudden jumps and changes of state in transient binary or physiological signals.
- Orthogonality: It guarantees perfect reconstruction of the signal without loss of information [21], [25].

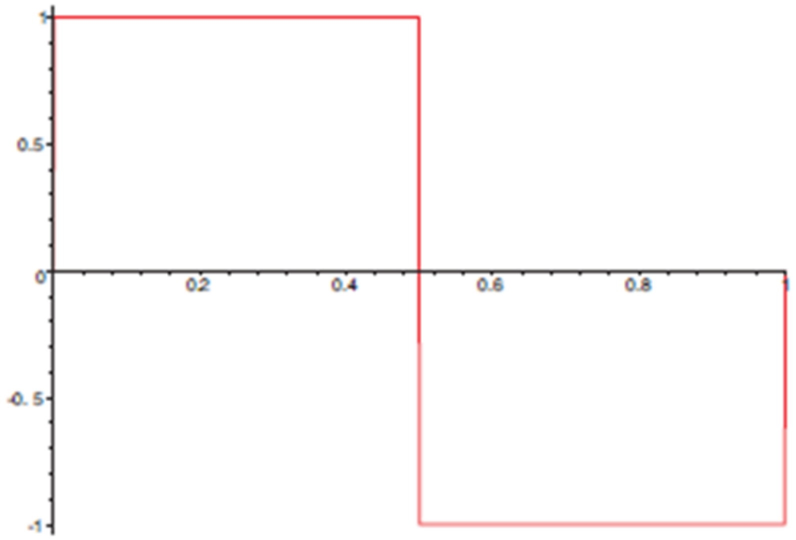
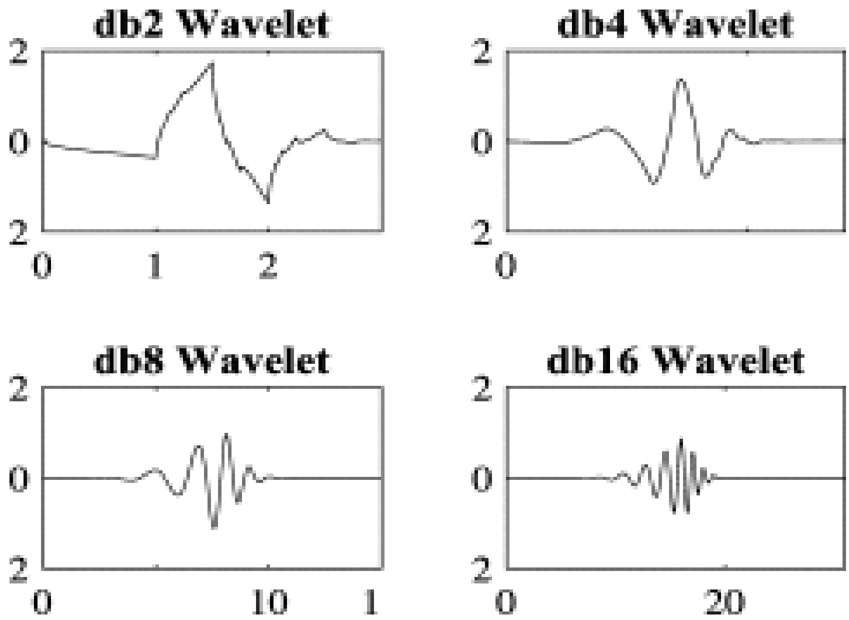


Figure 2.3: Representation of the Haar wavelet

As illustrated in Figure 2.4, the "stair-step" structure confirms its ability to act as a temporal contour detector. However, this lack of regularity limits its effectiveness for reconstructing slowly varying signals. This is why other, "smoother" families are often preferred for natural signals, hence the need to explore the Daubechies family [21], [25].

- **Daubechies family (dbN):**Daubechies wavelets (dbN) constitute a family of orthogonal wavelets with compact support (Figure 2.5). They are constructed iteratively in order to maximize the number of zero moments (N), which allows the signal energy to be concentrated on a small number of coefficients [21], [25].

They are parameterized by the order N of the vanishing moments. A higher order (such as db4 or db8) offers better regularity, making them particularly effective for denoising and data compression according to the JPEG2000 standard [16], [26].



[2]

Figure 2.4: Comparison of the shape and regularity of Daubechies wavelets (dbN) for orders N = 2, 4, 8 and 16

- **Symlets and Coiflets:**Symlets are a quasi-symmetric variant of Daubechies, designed specifically to minimize phase distortions (Figure 2.6). Coiflets, on the other hand, have zero moments for both the wavelet and the scaling function, making them ideal for fine approximation and derivative estimation [21].

Figure 2.5: Symmlet wavelet Sym8

2.3.2 Continuous Families (Continuous Analysis CWT):

Unlike orthogonal families, continuous wavelets often have infinite (theoretical) support while offering excellent time-frequency localization. They are particularly well-suited for phase extraction and fine-structure analysis in complex signals [16], [17].

- **Morlet's wavelet:** It is defined as a Gaussian modulated by a complex sinusoidal wave. Its structure gives it high frequency selectivity [22]:

$$\boldsymbol{\Psi}_{\boldsymbol{morlet}}\,(\boldsymbol{x}) = \quad \boldsymbol{e}^{\boldsymbol{jex}}\boldsymbol{e}^{\frac{-\,\boldsymbol{a}\,\boldsymbol{x}^{2}}{2}} \qquad \textit{\textbf{(2.5)}}$$

This wavelet is essential for the analysis of oscillatory signals, particularly in seismic, biomedical (EEG/ECG) and technical machine diagnostic fields [27].

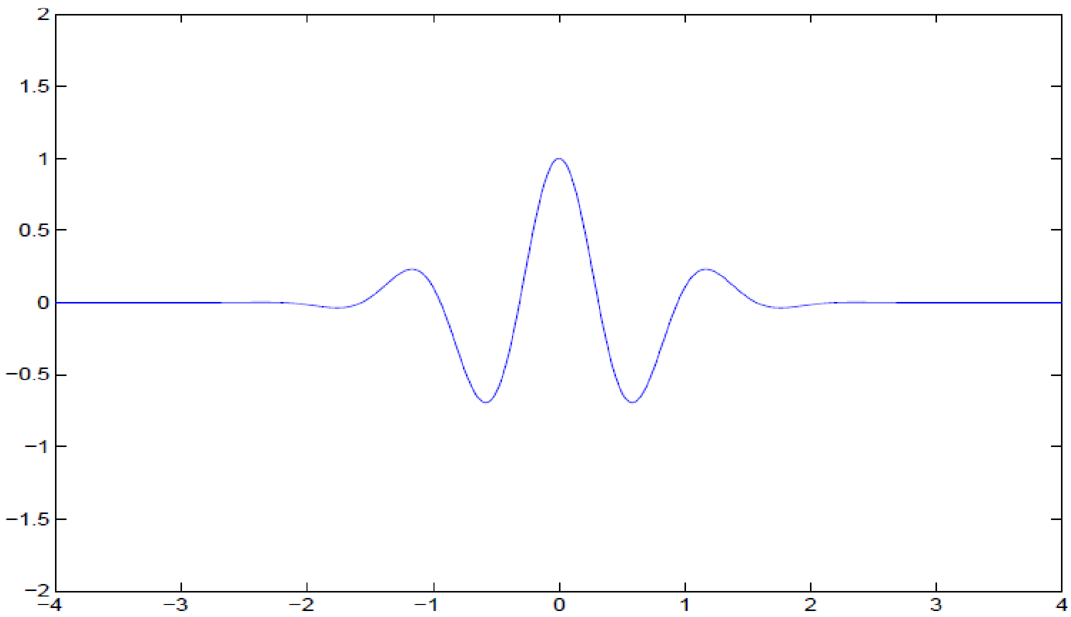


Figure 2.6: Structure of the Morlet wavelet

Analysis of Figure 2.7 reveals the modulation of the oscillation by the Gaussian envelope. This synergy allows for an optimal compromise between time and frequency resolution, which facilitates the detection of the instantaneous frequency of a non-stationary signal [23].

- **Mexican Hat:**Represented by the second derivative of a Gaussian function, it is a real-valued wavelet. Its perfect symmetry makes it ideal for peak detection. It is particularly effective for identifying transition fronts and for multi-scale analysis of fractal structures [16],[21].It is formulated as follows:

$$\Psi_{mexicain}(x) = (1 - 2x^2)e^{-x^2} \quad (2.6)$$

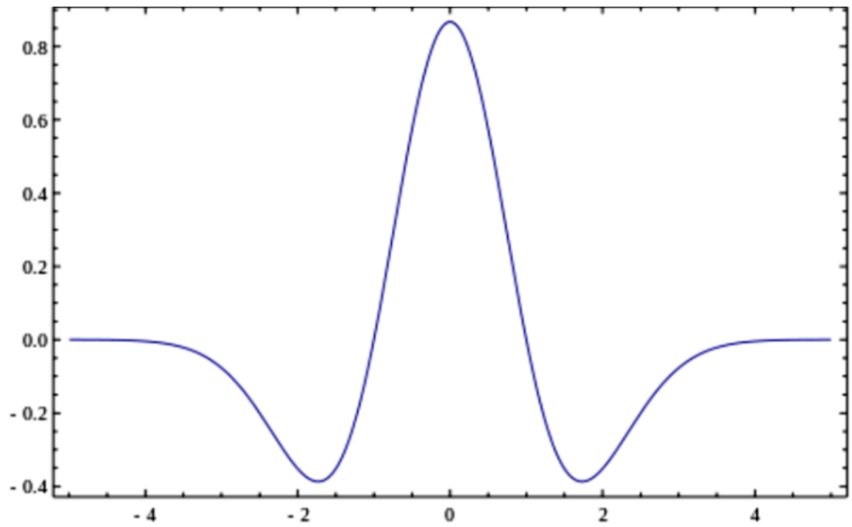


Figure 2.7: Time-domain representation of the Mexican Hat wavelet

Observation of Figure 2.8 reveals a symmetrical and smooth shape, which ensures accurate time localization without phase distortion. This property is crucial for detecting signal peaks and abrupt discontinuities with high fidelity [22] [27].

2.4. Discrete wavelet transformation (DWT):

The Discrete Wavelet Transform (DWT) is the digital alternative to the Continuous Wavelet Transform (CWT) and offers a multi-resolution framework for analyzing a signal. The DWT is based on the iterative decomposition of a signal by filtering with mirror high-pass and low-pass filters, defined by a mother wavelet.

This approach is characterized by its ability to precisely locate observable phenomena in terms of time-frequency while minimizing information redundancy. DWT has thus proven its effectiveness in detecting and capturing discontinuity and transient phenomena and has found its indispensable application in various fields such as data compression (notably JPEG 2000), denoising, and medical imaging [28] [29].

2.4.1 Introduction to the Discrete Wavelet Transform (DWT):

The Discrete Wavelet Transform (DWT) addresses the need for numerical computation and its algorithmic efficiency. While the CWT analyzes the signal redundantly over an infinite number of scale levels, the DWT processes the signal over a subset of levels and geometric positions, thus offering non-redundant multi-resolution processing [28][31].

2.4.2 Definition and Dyadic Diagram:

To eliminate the redundancy of CWT, DWT relies on discrete sampling of the scale (a) and translation (b) parameters. The most common choice is dyadic sampling (powers of 2):

- $a = 2^j$
- $b = k.2^j$

The simplified mathematical expression then becomes:

$$\psi_{j,k}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t-k.2^j}{2^j}\right) \qquad (2.7)$$

Where j represents the level of decomposition (scale) and k the time position [27].

2.4.3 Mallat's Algorithm: The Filter Bank

The practical implementation of DWT relies on Mallat's algorithm, which uses quadrature mirror digital filters. The signal is iteratively decomposed into two components:

- **Approximation Coefficients (A):**Obtained via a low-pass filter (h0), they capture the low frequencies and the general shape of the signal.
- **Detail Coefficients (D):**Obtained via a high-pass filter (h1), they isolate high frequencies, discontinuities and abrupt variations.

At each step, the signal is downsampled by a factor of 2, which allows the resolution to be divided by two while doubling the frequency precision[27][28].

2.4.4 Advantages of Discrete Decomposition:

DWT offers unique properties that often surpass classical analysis:

1. **Reducing redundancy:**Unlike CWT, DWT retains only the minimum number of coefficients necessary to perfectly reconstruct the signal.
2. **Spatio-frequency localization:**It accurately identifies the time of appearance of high frequencies (transients) while maintaining an overview of low frequencies.
3. **Orthogonality:**The use of orthogonal wavelets (such as Daubechies) facilitates compression and denoising without loss of information[30].

2.4.5 Applications:

Thanks to its iterative structure, DWT is the reference tool for:

- **Image compression:** Standard JPEG 2000.
- **Denoising:** Effective separation of noise (often located in the details) and the useful signal.
- **Edge detection:** In medical imaging and computer vision[27].

2.5 Dyadic Wavelet Transform:

The Dyadic Wavelet Transform is used as a hybrid compromise between continuous CWT and standard discrete DWT. These two transforms have been more commonly used for scale and time sampling, respectively. The dyadic transform attempts to shift sampling to scale and maintain continuous sampling over time in signal analysis. While in DWT both the scale and delta t are sampled, in this transform only the scale is sampled [27].

The Dyadic Wavelet Transform is a form of strategic compromise between the Continuous (CWT) approach and the Discrete Standard (DWT) approach, and it is particularly used when it is necessary to retain the Discrete approach for numerical computation and the Continuous approach for accurate time-domain signal analysis [30].

Chapter 3: Multiresolution Analysis and Applications

3.1 Introduction:

Multiresolution analysis (MRA) provides the theoretical and algorithmic foundation for transforming wavelet theory into a powerful and fast computational tool for digital signal processing. Unlike classical approaches such as the Fourier transform, which impose a fixed resolution, MRA relies on an adaptive decomposition offering an optimal compromise between time and frequency resolution. This property is fundamental for the analysis of non-stationary signals where transient phenomena require multi-scale observation. [32] [16]

Formally, this analysis relies on the concept of nested subspaces, where each level j represents an approximation of the signal at a given resolution. The transition between these scales is ensured by two key functions: the scaling function for the approximations and the mother wavelet for extracting details. [16] $\{\boldsymbol{V_j}\}_{\boldsymbol{j}\in\mathbb{Z}}\boldsymbol{L^2}(\mathbb{R})\boldsymbol{\Phi}(\boldsymbol{t})\boldsymbol{\Psi}(\boldsymbol{t})$

The aim of this chapter is to detail the implementation mechanisms of this theory. We will first discuss multirate filtering, a technological pillar of decimation and interpolation, necessary for changing the sampling frequency. Next, we will analyze quadrature mirror filter (CQF) structures that allow for perfect signal reconstruction [32]. These theoretical foundations, extensively documented by industrial tools such as MathWorks, will serve as the basis for the experimental implementation under GNU Radio presented later in this work.

3.2 Multirate Filtering: Mathematical Analysis and Structures

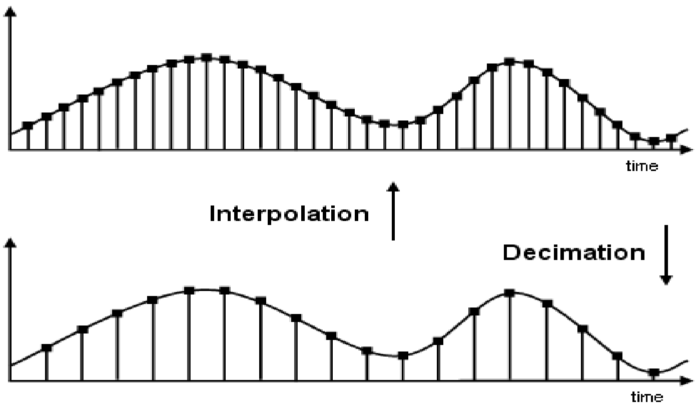


Figure 3.1: Illustration of the principle of interpolation and decimation in the time domain [31]

Multirate filtering is a key technological building block in multiresolution MRA analysis and Software-Defined Radio (SDR) architectures. Its main role is to adapt the sampling rate to the bandwidth of the processed signal in order to optimize the computational load (complexity reduction) [31]. This technique relies on two fundamental dual operators: decimation and interpolation (Figure 3.1).$\boldsymbol{f}_s$

3.2.1 Decimation (Downsampling) Process:

Decimation consists of reducing the sampling frequency by an integer factor M. Conceptually, an efficient decimator is not limited to simple sample removal, but necessarily incorporates a prior filtering step.

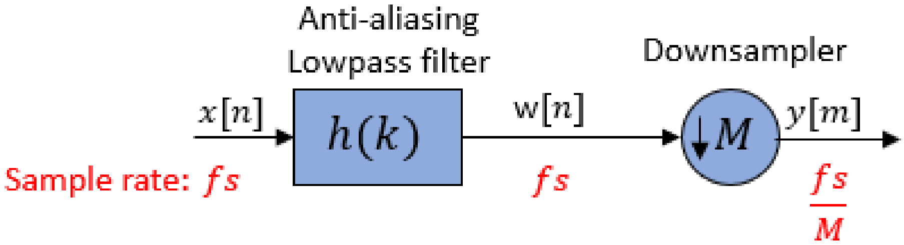


Figure 3.2: Block diagram of a decimation system including an anti-aliasing filter [32]

As illustrated by the modeling [32], the input signal x(n) first passes through a digital low-pass filter h(k), described as an anti-aliasing filter.
The role of this filter is to limit the signal bandwidth to the Nyquist frequency of the new data rate, i.e., $f_s/2M$, in order to avoid spectral aliasing [33]. The time relationship between the input signal x[n] and the decimated signal y[n] is expressed by: f_s

$$y(n) = x(nM) \tag{3.1}$$

In the frequency domain, to avoid any loss of information, the spectrum of the resulting signal is defined by the summation of the shifted spectral replicas:

$$Y(z) = \frac{1}{M}\sum_{m=0}^{M-1} X(z^{\frac{1}{M}}\, e^{-j\frac{2\pi m}{M}}) \tag{3.2}$$

Figure 3.3 shows an example of a decimation system with M = 3. As illustrated in this figure, the 6 kHz input signal is first filtered to eliminate any frequency component above 1 kHz (f_s/2M). Then, the downsampling block reduces the bitrate by a factor of M=3, producing a 2 kHz output signal without any aliasing-related distortion. [32]f_s

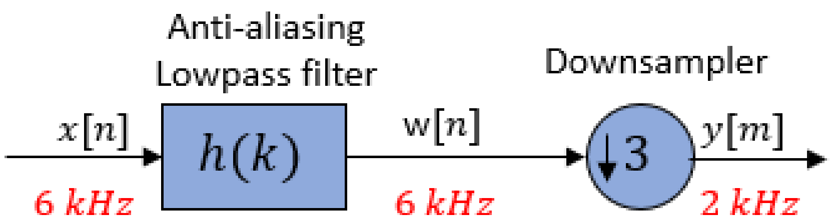


Figure 3.3: Example of application of a decimator with an input frequency of 6 kHz [32]

This mathematical relationship [33] is illustrated by the example in Figure 3.4, where we observe the effect of decimation on the signal spectrum as well as the reduction in its rate.

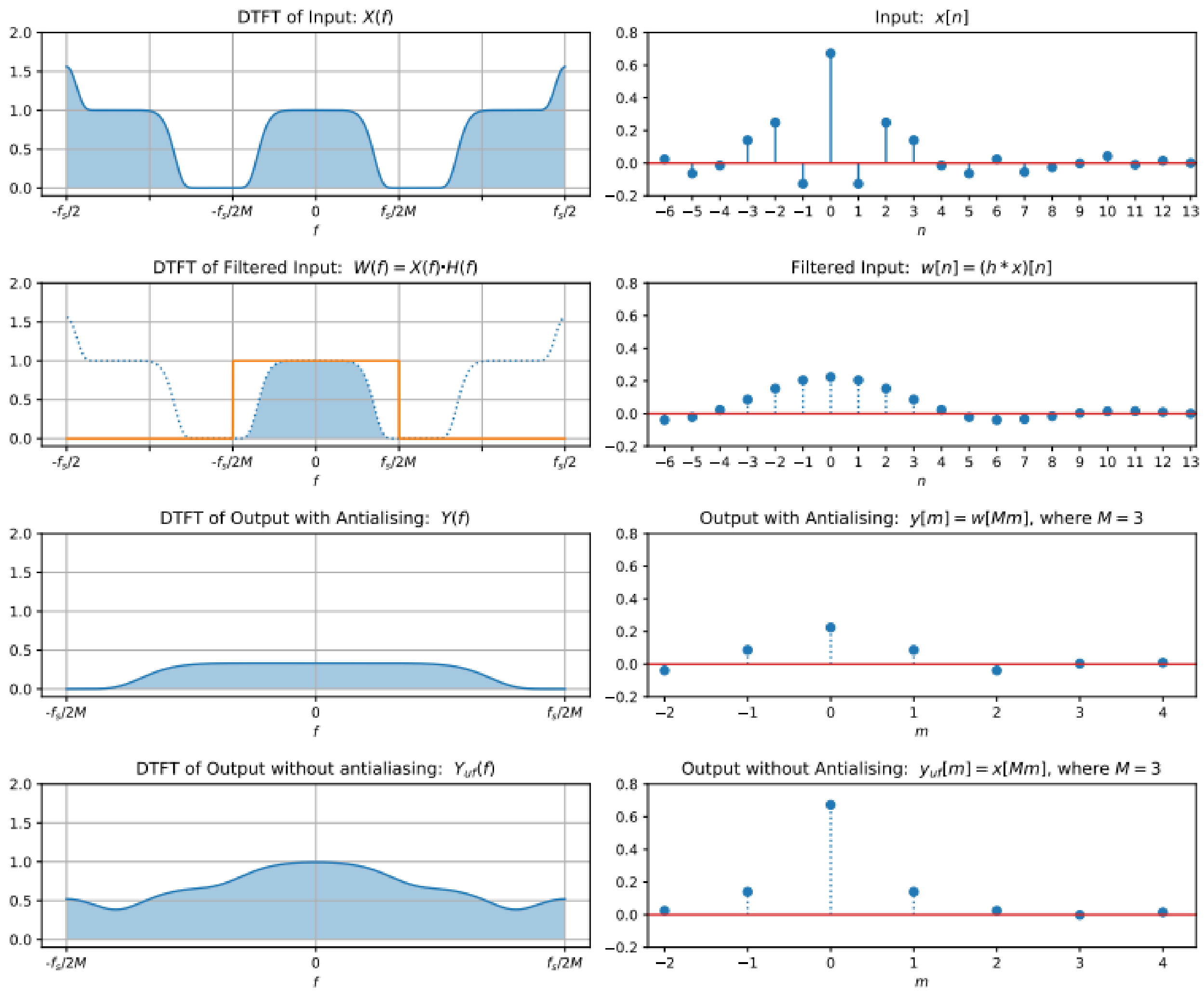


Figure 3.4: Spectral (DTFT) and temporal analysis of the decimation process with and without anti-aliasing filtering[32]

The spectral analysis presented in Figure 3.4 highlights the crucial importance of prior filtering. It can be observed that without an anti-aliasing filter, spectral replicas overlap (aliasing), which distorts the original information. In contrast, applying the *h(k)* filter limits the frequency content to the allowed band, thus ensuring perfect decimation and faithful signal reconstruction. [33]

3.2.2. Interpolation Process (Upsampling):

Conversely, interpolation aims to increase the sampling frequency by a factor of L [31]. This process begins with the upsampling unit (L) which inserts L-1 zero-value samples between each original sample, such as:↑

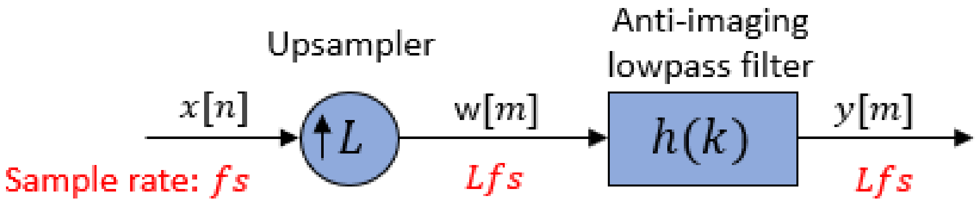


Figure 3.5: Structure of a digital interpolator (Upsampler followed by an anti-imaging filter) [39]

$$v(n) = \begin{cases} x(\frac{n}{L}) & \textbf{\textit{si n est un multiple de L}} \\ 0 & \textbf{\textit{sinon}} \end{cases} \qquad \textbf{(3.3)} \qquad \textbf{[31]}$$

Although this operation formally increases the throughput, it generates unwanted spectral components called images. To finalize the operation, a low-pass filter, known as an anti-imaging filter, is applied in cascade.
In order to remove these replicas and smooth the signal, the spectrum of the interpolated signal V(is then related to the original spectrum by the following equation:e^{jLw})

$$\mathbf{V}(e^{jLw}) = X(e^{jLw}) \qquad \textbf{(3.4)} \qquad \textbf{[32]}$$

Figure 3.7 shows an example of an interpolation system with L = 3.

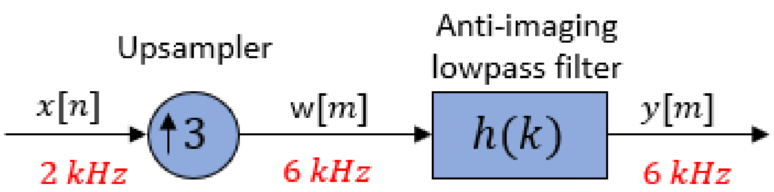


Figure 3.6: Numerical example of an interpolation with a factor L=3 [32]

This example concretely illustrates the increase in sampling frequency. The original signal, clocked at 2 kHz, is tripled by the upsampling unit (L=3) to reach 6 kHz. Note that the insertion of zeros generates spectral components (images) which the low-pass filter h(k) (anti-imaging) eliminates [32]. The final result is a 6 kHz signal with a smoothed time-domain shape that faithfully reproduces the original information. [32]

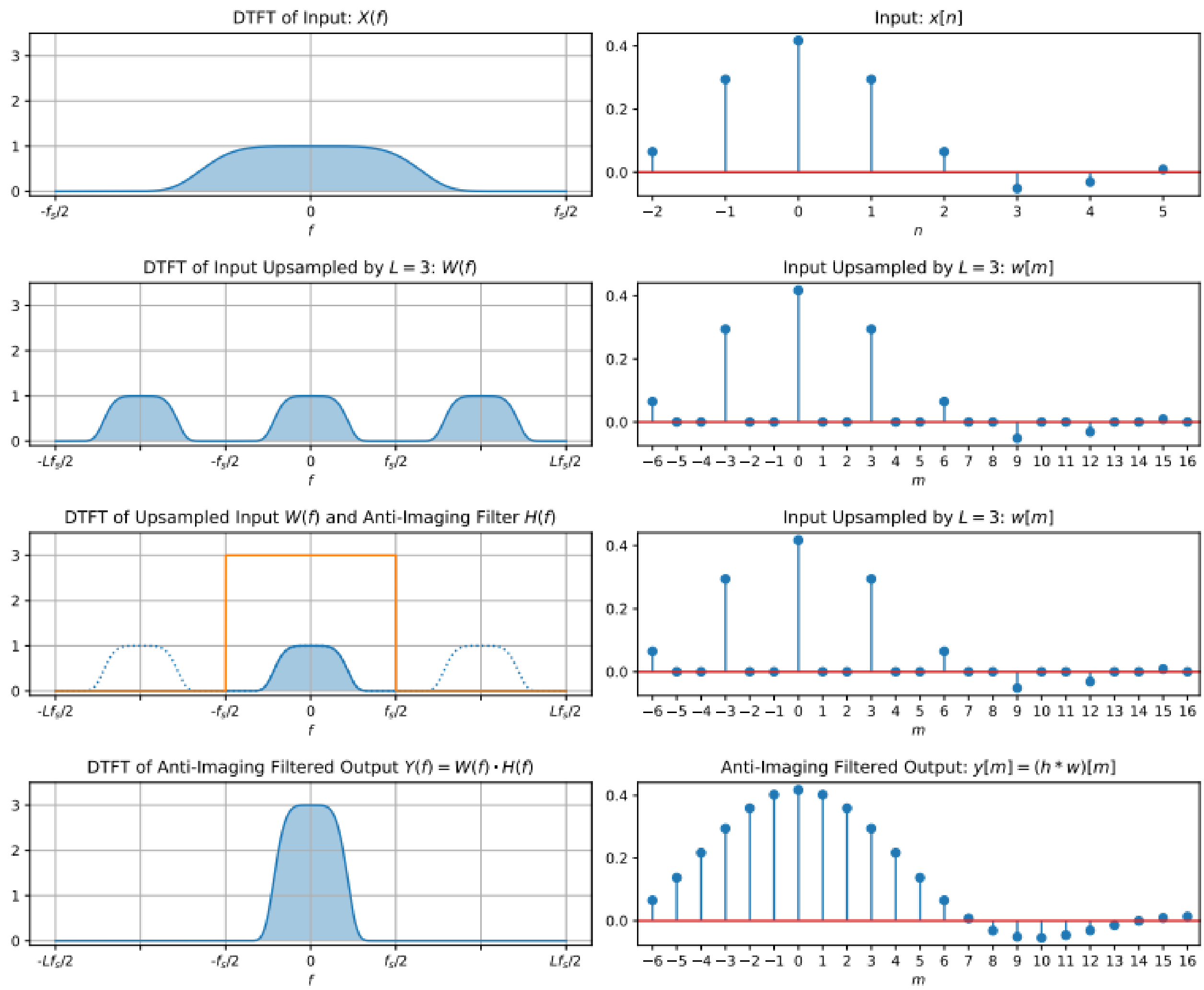


Figure 3.7: Spectral (DTFT) and temporal analysis of the interpolation process (effect of zero insertion and anti-imaging filtering) [32]

It is observed that increasing the data rate by inserting zeros creates spectral replicas called images. The intervention of the low-pass filter H(f) is then essential to remove these unwanted components. In the time domain, this filtering allows the missing samples to be reconstructed, thus transforming the pulse sequence into a smooth and continuous signal.

3.2.3. Sampling Frequency Conversion

In practical applications, it is common to have to convert the sampling rate by a rational factor I = where L and M are integers (for example, the conversion from 44.1 kHz to 48 kHz). This process requires a hybrid architecture called a "Sample Rate Converter" [33].$\frac{L}{M}$

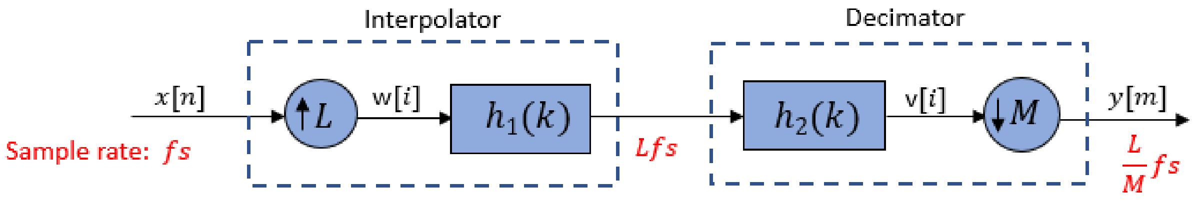


Figure 3.8: Cascade architecture of a rational factor frequency converter$\frac{L}{M}$[31]

To perform this operation without signal degradation, interpolation is systematically carried out using L followed by decimation using M. This ordering choice is crucial. Increasing the bitrate first preserves the entire original spectral content before applying decimation.

In terms of implementation, instead of using two separate filters (Anti-imaging and Anti-aliasing), a single low-pass filter (n) is designed and placed between the upsampler and the downsampler.h_{LP}

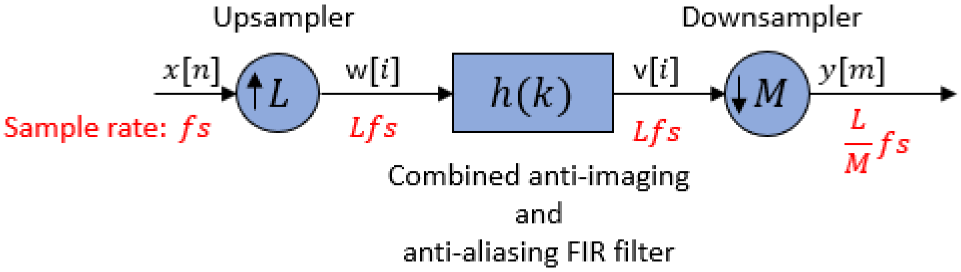


Figure 3.9: Optimized structure of a frequency converter with a common filter [32]

The implementation of this optimized structure (Figure 3.9) relies on the design of a hybrid filter capable of providing dual protection. On the one hand, it must suppress spectral images generated by interpolation (L). On the other hand, it must limit the signal bandwidth to prevent aliasing before decimation (M). Consequently, its ideal cutoff frequency is determined by the most restrictive constraint:ω_c

$$\omega_c = (\frac{\pi}{L}\frac{\pi}{M}) \qquad (3.5) \qquad [31]$$

Figure 3.10 illustrates this process concretely. It considers the conversion of a 2 kHz signal to 3 kHz. The operation takes place in two key steps: first, the upsampling unit increases the bitrate by a factor of L=3, raising the frequency to 6 kHz. At this stage, the combined filter eliminates spectral images while preparing the signal for reduction. Finally, the decimation block divides the frequency by M=2 to achieve the target bitrate of 3 kHz. This approach allows the sampling frequency to be changed precisely without introducing aliasing. [32]

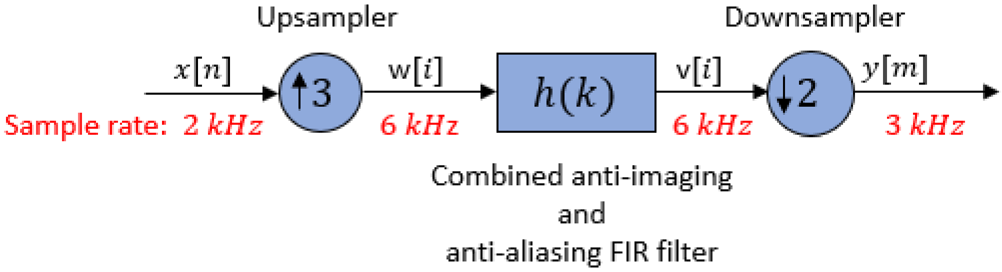


Figure 3.10: Example of frequency conversion for a rational factor I =$\frac{r}{l}$[32]

Figure 3.11 provides an overview of the frequency conversion with L=3 and M=2. In the frequency domain (left), the effectiveness of the hybrid filter [40] (outlined in purple) is evident, isolating the useful spectrum while eliminating spectral artifacts due to interpolation. This filtering also ensures the absence of aliasing before the decimation phase. Meanwhile, the time domain (right) illustrates how the intermediate signal v(i) is smoothed by adding interpolated samples before being downsampled to obtain the final signal y(m) at the target frequency. [32]

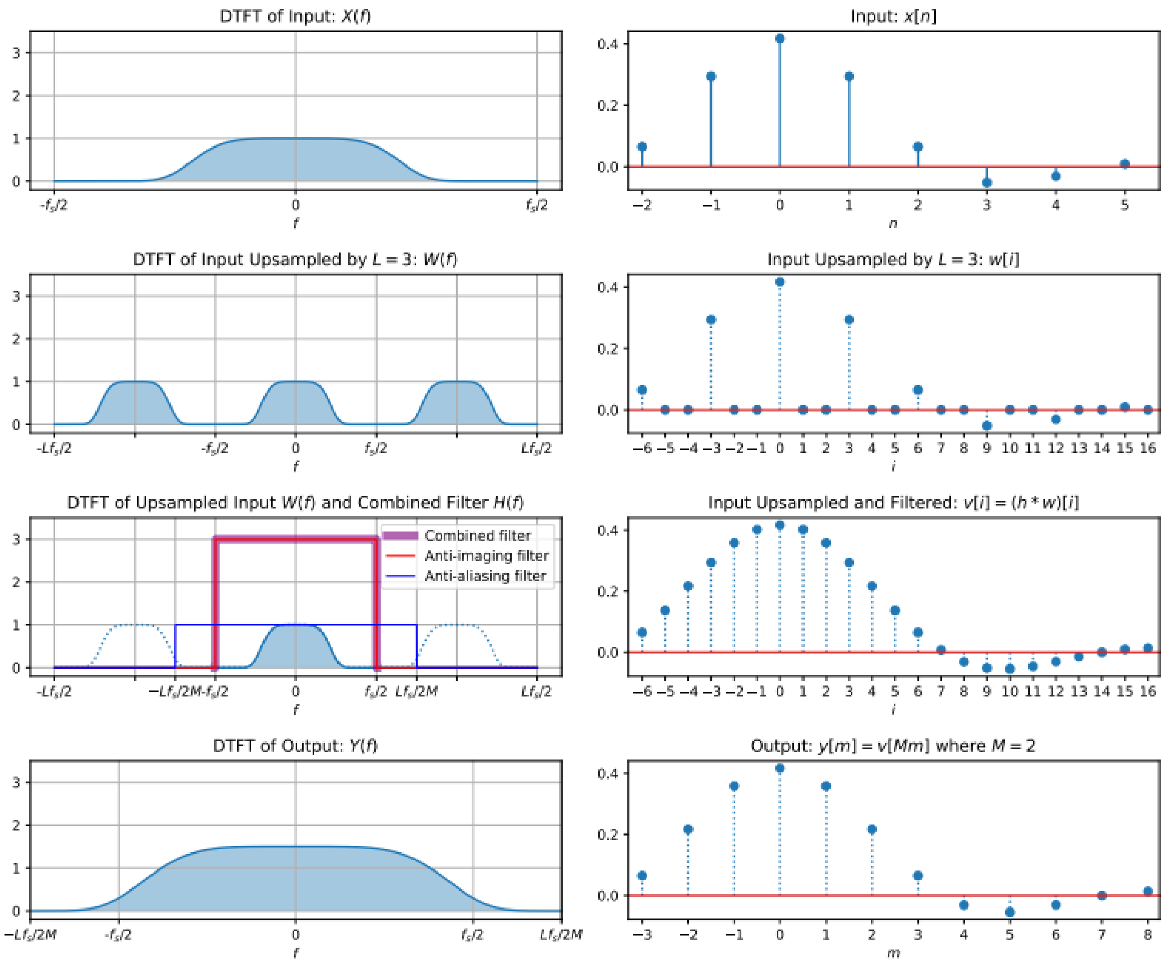


Figure 3.11: Spectral (DTFT) and time analysis of the frequency conversion process by a rational factor 3/2 [32]

3.3 QMF and CQF Filters

3.3.1 Analysis of Quadrature Mirror Filters (QMF) and aliasing cancellation

The structure of a 2-channel filter bank (Analysis and Synthesis) is illustrated in the following Figure 3.12:

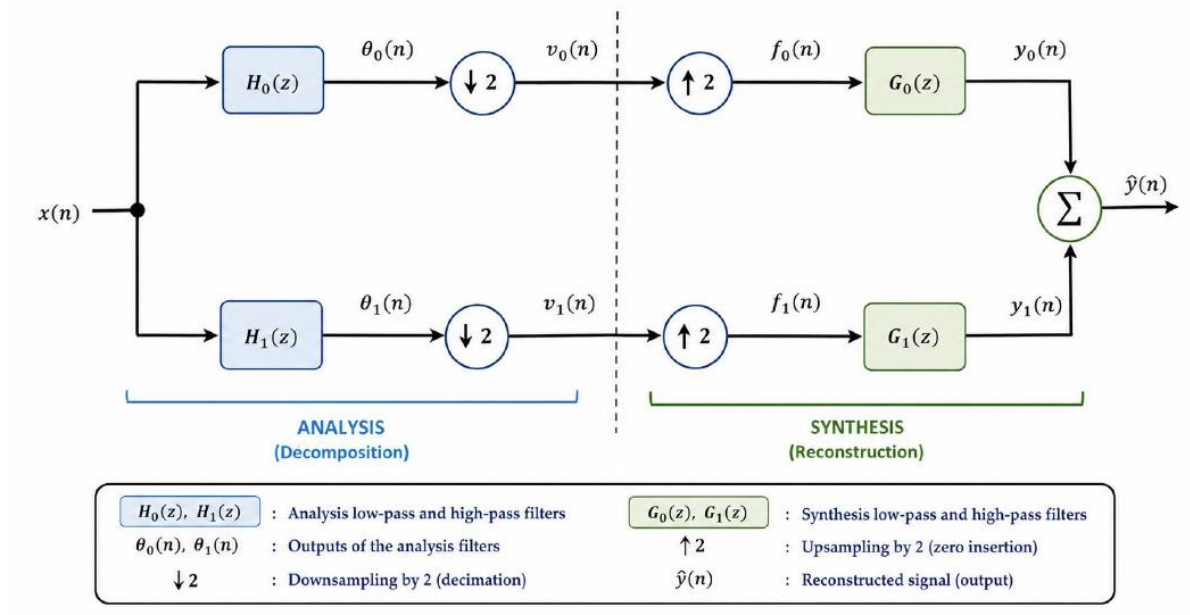


Figure 3.12: Block diagram of a 2-channel filter bank [32]

The fundamental problem with two-channel filter banks lies in the spectral aliasing phenomenon generated by the decimation step. To obtain a coherent reconstruction, the output signal Y(z) is modeled by the global transfer function.

$$\boldsymbol{Y(z) = \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)]X(z) + \frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)]X(-z)} \qquad \textbf{(3.6)}$$

The cancellation of the aliasing component (the term en) is guaranteed by imposing the following mirroring conditions [33]:$X(-z)$

$$\boldsymbol{H_1(z) = H_0(-z)\,, G_0(z) = H_0(z)\,, \qquad G_1(z) = -H_1(z)} \qquad \textbf{(3.7) [33]}$$

This structure allows the system to be reduced to a simple amplitude distortion. In practice, the implementation is optimized via polyphase networks, which allows signals to be processed at the reduced Nyquist frequency, thus minimizing algorithmic complexity.[31].

3.3.2 Conjugate Quadrature Filters (CQF) and Perfect Reconstruction (PR) condition:

Although QMF filters address aliasing, they can introduce significant amplitude distortion. CQF (Conjugate Quadrature Filters) overcome this limitation byintroducing a strict orthogonality condition, essential for a Perfect Reconstruction (PR).

In accordance with the work of A. Cohen [41], this orthogonality translates into respecting the identity of power in the frequency domain:

$$\left|H_0\,(e^{j\omega})\right|^2 + \left|H_0\,(e^{j\omega+\pi})\right|^2 = 2 \qquad \textbf{(3.8)}$$

Mathematically, this constraint ensures that the impulse responses are orthogonally shifted ( ), which allows for full conservation of the signal energy (Parseval's property). It is this rigorous framework that allows the generation of orthonormal wavelet bases [38].

$$\sum h_0(n)h_0(n-2k) = \delta(k) \qquad \textbf{(3.9)}$$

3.3.3 Implementation and convergence towards multiresolution analysis

The interest of CQF filters goes beyond simple digital filtering to constitute the core of multiresolution analysis (MRA). The link between these filters and Daubechies wavelets is established by the regularity of the scaling filter (z). To ensure convergence to a stable scaling function (x), the filter must have a number K of zeros at the frequency $f_s/2$ (Z = 1), which translates into the following formulation [38]:

$$H_0(z) = (1 + z^{-1})^K L(z) \qquad \textbf{(3.10)}$$

This formulation, the cornerstone of Mallat's algorithm, allows for decomposition into cascades (pyramidal structure). In multiresolution analysis, this choice guarantees a precise separation of details and approximations [32] [38], offering optimal time-frequency analysis for non-stationary signals.

3.4 Multiresolution analysis:

3.4.1 Definition:

Multiresolution analysis (MRA) is a fundamental signal processing technique that allows a signal to be represented at different levels of resolution or scale. Unlike classical methods such as the Fourier transform, it allows simultaneous analysis in the time and frequency domains (scale).

The idea is to decompose a signal into several parts, each corresponding to a particular piece of information according to a given scale. This approach is particularly suitable for the analysis of non-stationary signals [16].

3.4.2 Principle of decomposition:

Multiresolution analysis involves decomposing the signal into two types of components:

Approximation (A): concerns low frequencies (global signal information).

Detail (D): it corresponds to high frequencies (rapid variations).

At each level of decomposition, the signal is separated into two parts using filters:

- a low-pass filter
- a high-pass filter

A decimation operation (sampling reduction) is then applied to reduce the size of the data [16].

3.4.3 Multiresolution analysis (MRA):

It is based on two main functions:

Scaling function φ(t):

The scaling function φ generates approximations at a given resolution via orthogonal projections. It is often linked to a low-pass filter and satisfies a two-scale equation.

$$\varphi(t/2) = \sum h0(k)\, \varphi(2(t - k)) \qquad \textbf{(3.11)}$$

where h(k) are the filter coefficients. In AMR analysis, the approximation coefficients

$$aj(k) = <s(t), \varphi\{j,k\}(t)> \qquad \textbf{(3.12)}$$

capture the smoothed information from the signal [35] [16].

Wavelet function ψ(t):

The wavelet function ψ, orthogonal to φ, captures the details lost between resolutions with

$$\psi(t/2) = \sum h1(k)\, \varphi(2(t - k)) \qquad \textbf{(3.13)}$$

with h1 of the high-pass filter. The coefficients of detail

$$dj(k) = <s(t), \psi\_\{j,k\}(t)> \qquad \textbf{(3.14)}$$

extract local variations, ideal for non-stationary signals [16].

3.5 Practical problems of multiresolution analysis:

Multiresolution analysis (MRA) is the theoretical basis of discrete wavelet transform (DWT). It allows a signal to be represented at different scales by a hierarchical decomposition into nested subspaces.

AMR offers many advantages for signal and image processing, but it also has several practical limitations that affect the accuracy and stability of the results. These problems arise mainly during digital implementation [16].

3.5.1 Boundary Effects:

In multiresolution analysis, edge effects refer to all the distortions introduced when applying the wavelet transform to signals with finite support. In practice, the signals processed are necessarily limited in time or space, whereas the theoretical formulation of wavelets generally relies on functions defined over infinite domains.

The mismatch between the theoretical model and practical requirements gives rise to distortions at the boundaries of the signal, which impairs the reliability of the wavelet coefficients in these areas [36][16].

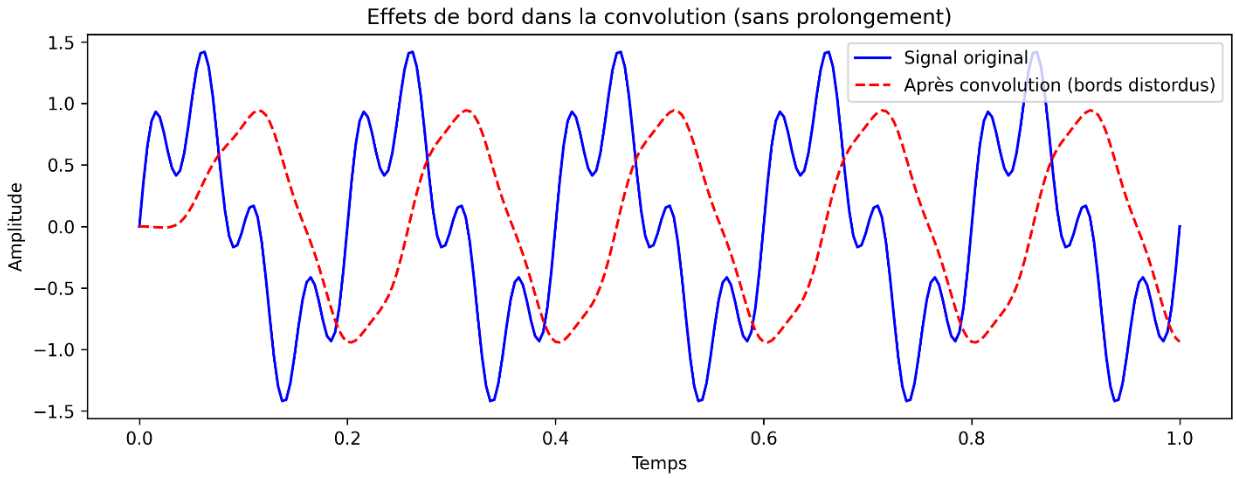


Figure 3.13 Side effects in convolution: comparison between the original signal and the convolved signal without extension[39]

The DWT relies on convolution plus decimation: for the approximation at the j-th level,

$$\boldsymbol{aj(k)=\sum h(m-2k)\ s(m)} \qquad \textbf{(3.15)}$$

where is the impulse response of the low-scale low-pass filter. For a finite signal, , the filter overflows at the edges:$h = \{h_n\}s(n)n = 0, \ldots, N-1$

LEFT: → requires extension$m < 0 s(-m)$

RIGHT :$m \geq N \rightarrow s[N+m]$[39]

3.5.2 Solutions:

Periodization: Cyclic extension of the signal, valid if periodic, but introduces artificial discontinuities (non-periodic edges).

Symmetry: Symmetric padding for symmetrical Daubechies; preserves orthogonality but artificially increases length.

Zero padding: Simple, but can introduce edge artifacts [39].

3.6 Correction and improvement techniques:

3.6.1 Signal Extension Methods: Zero Padding

Zero padding is the simplest and fastest method for handling edge effects in multiresolution analysis, consisting of adding zeros to the ends of the signal before the DWT convolutions. While appealing due to its algorithmic simplicity, it introduces systematic artifacts that often make it unsuitable for precise applications.

Zero-padding extends the original signal x(n) (defined for n = 0, ……, N-1) by setting x(n) = 0 for n < 0 and n > N. The number of added zeros must be at least equal to the FIR filter length minus one (M-1). [39]

3.6.2 Edge management by signal extension:

Edge management by signal extension in multiresolution analysis consists of artificially extending the finite signal x[n] (n=0...N-1) at the ends to allow complete convolutions of the wavelet filters $h_0(n)$ and $h_1(n)$ of length L without truncation,

Symmetrical (mirror image): $x[-n] = x[n]$ for $n = 1 \ldots L - 1$

Periodic: , without discontinuity for quasi-cyclic signals $x[n + N] = x[n]$ [39].

3.6.3 Overlap methods (Overlap-Add / Overlap-Save):

The Overlap method resolves block discontinuities by processing the signal in overlapping blocks, eliminating artificial jumps at block boundaries while restoring local translation invariance in the DWT. It is well suited for Wavelet Packets (WP) and long sequences [33].

3.7 Conclusion:

This chapter has exhaustively demonstrated that multiresolution wavelet analysis, despite its identified fundamental practical challenges, becomes a robust tool thanks to advanced corrective techniques of (zero-padding, symmetric extensions, overlap-add/save)[16][39].

Chapter 4: Implementations

4.1. Introduction to GNU Radio

In this phase of our study, the choice of implementation software is essential to validate the multiresolution analysis algorithm (MRA). We worked with GNU Radio, an open-source development framework specializing in digital signal processing (DSP) and software-defined radio (SDR).

The architecture of this software is based on a modular approach: Signal processing is divided into several units called "blocks". These are linked together to create a Flowgraph (flow diagram), which allowed us to manage the passage of data through the stages of our analysis.

The use of GNU Radio in this chapter is justified by several technical points:

- **Customization via Python:**The ability to code specific blocks was crucial for integrating our own Discrete Wavelet Transform (DWT) algorithm, as standard blocks were insufficient to cover our filtering needs.
- **The GRC (GNU Radio Companion) interface:**This tool served as our visual interface for building the processing chain, linking the blocks (sources, filters, etc.) and automatically generating the execution code.
- **Source flexibility:**This environment allowed us to test the AMR analyzer on simulated signals (such as the Cosine signal) as well as on real signals captured by SDR hardware (RTL-SDR key).

In summary, GNU Radio allowed us to move from theory to practical experimentation, observing the results that we will detail in the following sections.

4.2. Block AMR analysis of a Cosine signal

In this first experiment, we apply multiresolution analysis (MRA) to a simple deterministic signal (Cosine). The objective is to validate the proper functioning of our MRA block and to observe the behavior of the signal after processing.

4.2.1. Block diagram of the analyzer

The processing follows a well-defined modular chain. The following diagram illustrates the signal processing steps, from generation to display :

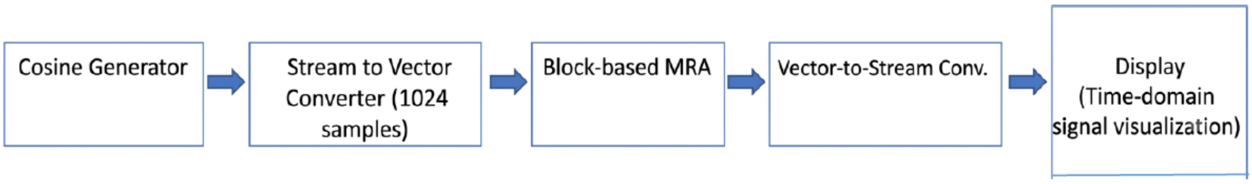


Figure 4.1: Block diagram of the AMR analyzer for a cosine signal

- **Cosine generator:**Produces the deterministic source signal at a given frequency (1 kHz).
- **Stream to Vector Converter:**Groups continuous samples into blocks of 1024 to prepare for vector processing.
- **Block-Based MRA :** Performs 3-level multiresolution decomposition (DWT) based on CQF (Conjugate Quadrature Filters) on each block.
- **Vector to Stream Converter:**Reconverts data blocks into a continuous stream of samples.
- **Display :**Allows for comparative time visualization of the original signal and the processed signal.

4.2.2. Flowgraph GNU Radio and Simulation

The corresponding flowgraph was implemented under the GRC interface. Figure 4.2 shows the organization of the blocks and the chosen sampling parameters (48 kHz).

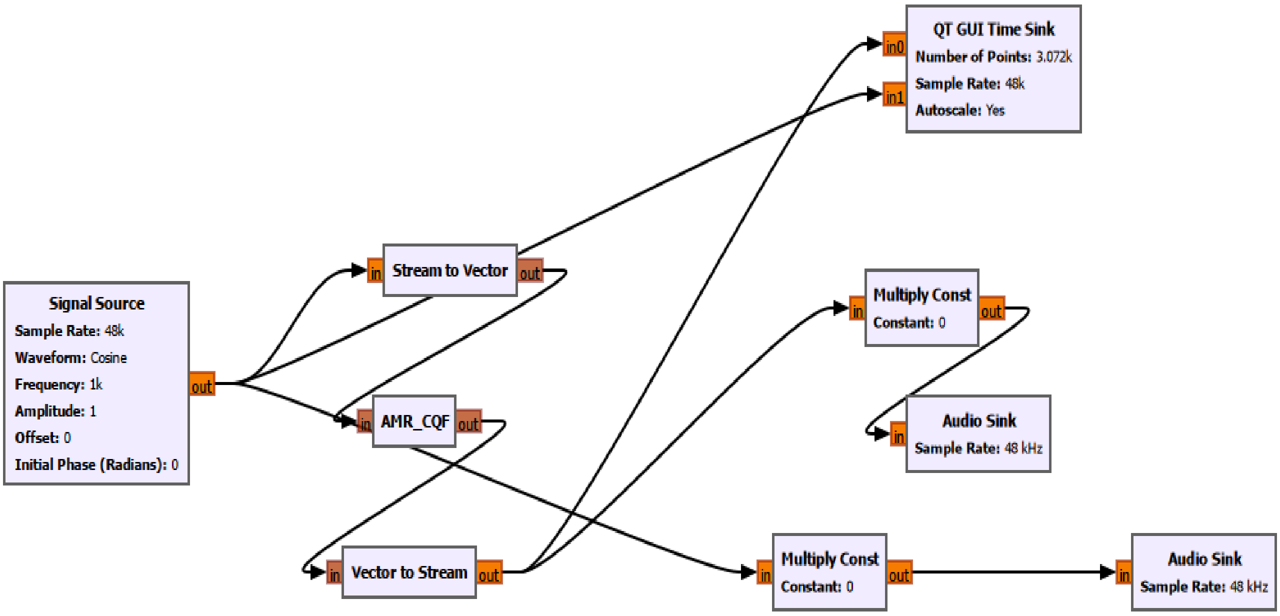


Figure 4.2: Implementation of the AMR analysis flowgraph for a cosine signal under GRC

4.2.3. Analysis of the edge effect problem

After running the real-time simulation, we obtain the time-domain results displayed on the virtual oscilloscope (QT GUI Time Sink) shown in Figure 4.3

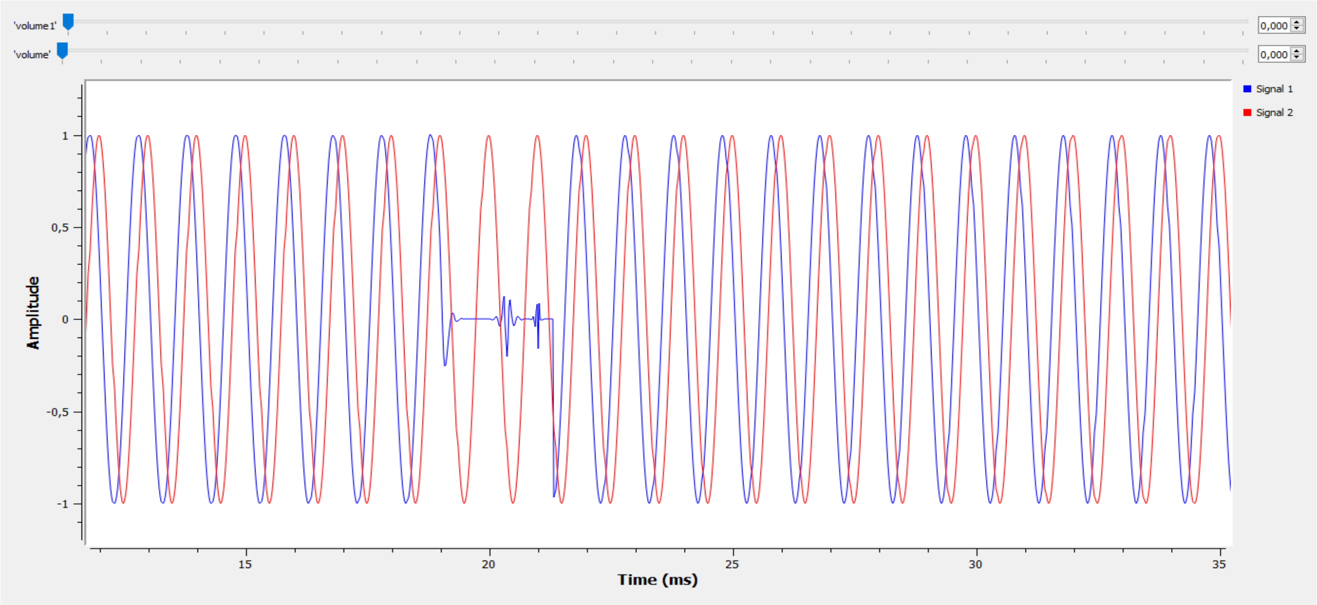


Figure 4.3: Results of the time-domain simulation of the processed cosine signal and the edge effect problem

According to the results in Figure 4.3, disturbances are clearly visible at the edges of each block. While the original signal (in red) retains a perfect sinusoidal shape, the signal processed by the MRA (in blue) undergoes significant degradation at the beginning and end of the block (1024 samples), where the amplitude drops sharply to zero with noisy oscillations. This phenomenon, known as edge effect, degrades the continuity of the processed signal.

From a technical standpoint, this is explained by the fact that the filtering process operates on isolated segments. At the beginning and end of each 1024-sample window, the algorithm lacks the necessary neighboring values (past and future samples) for correct boundary calculations. To address this issue, we subsequently introduced the Zero-Padding and Overlap methods.

4.3. Block MRA analysis of an SDR signal

In this second step, we replace the deterministic signal with a physical signal captured via the RTL-SDR hardware interface. This procedure allows us to verify the robustness and behavior of our MRA block.

4.3.1. Flowgraph and Configuration

For this step, we configured a real source using the RTL-SDR Source block under GRC. As shown in the flowgraph, the center frequency is set to 94.5 MHz (commercial FM band) with an initial sampling rate of 1.6 MSps.

The raw captured stream first undergoes low-pass filtering, then passes through the WBFM Receive block with an audio decimation of 50. The resulting audio signal, brought down to a suitable sampling frequency of 32 kHz, is then converted into a vector before being injected into our AMR_CQF multiresolution analyzer.

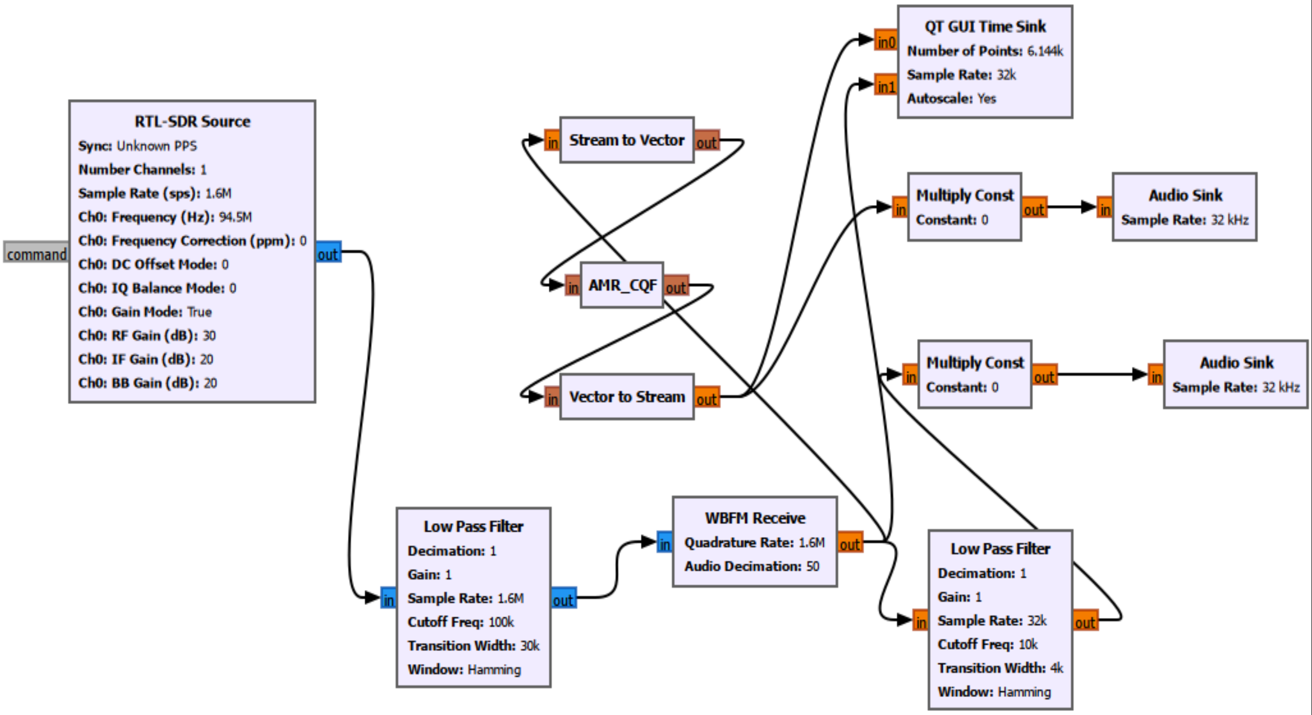


Figure 4.4: Implementation of the GRC flowgraph for processing a real FM signal via SDR

4.3.2. Results and Observations

By running this flowgraph, we obtain the demodulated and decomposed audio signal in real time on the virtual oscilloscope (QT GUI Time Sink), shown in Figure 4.5. The edge effect results in annoying noise on the demodulated signal.

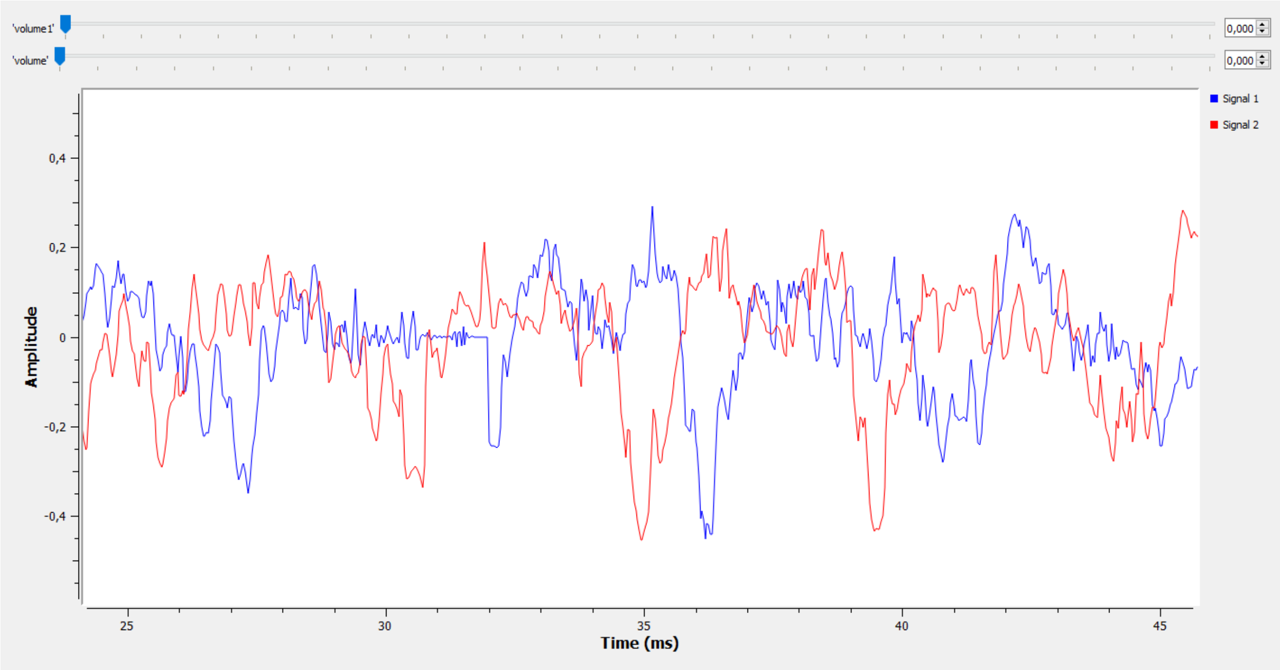


Figure 4.5: Temporal visualization of the edge effect on a real SDR signal

Visual analysis of the curves in Figure 4.5 clearly shows that the edge effect phenomenon manifests itself even more critically on a real signal:

- The original demodulated signal (in red) follows normal dynamic and random variations, characteristic of an audio signal from a real FM capture.
- The processed signal (in blue) undergoes a sharp and abrupt cut-off around t = 32 ms. At this precise moment, the signal amplitude collapses completely and freezes at zero in the form of an artificial flat plateau, completely breaking the continuity of the original signal.

These observations definitively confirm that edge effects are an intrinsic problem with block segmentation and finite filtering. They are completely independent of the nature of the source (whether simulated deterministic like the Cosine function, or real like the SDR signal). This blatant degradation fully justifies the need to implement correction techniques to ensure temporal continuity.

4.4. Treatment of edge effects using the Zero Padding method (Cosine case)

One of the first solutions explored to mitigate edge effects is the addition of zeros, a technique called ZeroPadding. The idea is to pad each block of data with null samples (zeros) before the filtering step in order to give the filter the values needed to calculate the limits without abruptly distorting the useful signal.

4.4.1. Principle and Implementation

The implementation of this method under GRC relies on integrating the padding mechanism directly into our custom block. The signal from the Source Signal block (Cosine at 1 kHz) is converted into vectors, then padded with zeros before undergoing multiresolution analysis.

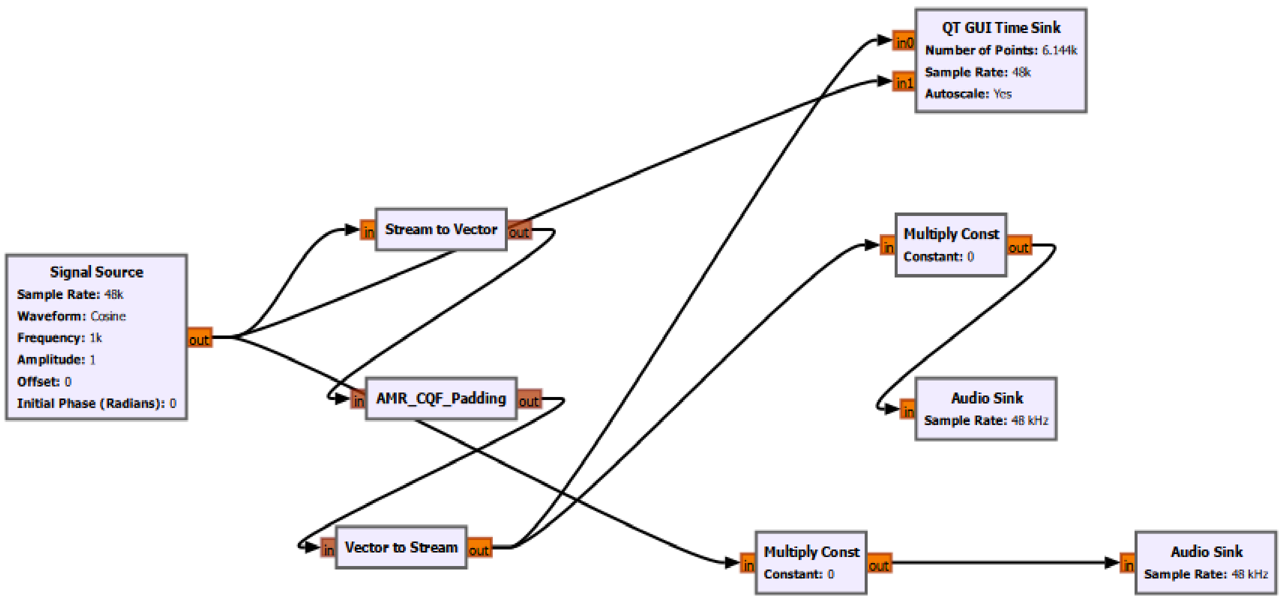


Figure 4.6: GRC flowgraph incorporating the ZeroPadding method for the Cosine signal

4.4.2. Results and Observations

Executing the flowgraph with Zero Padding allows us to obtain the time-based results on the block (QT GUI Time Sink), shown in Figure 4.7.

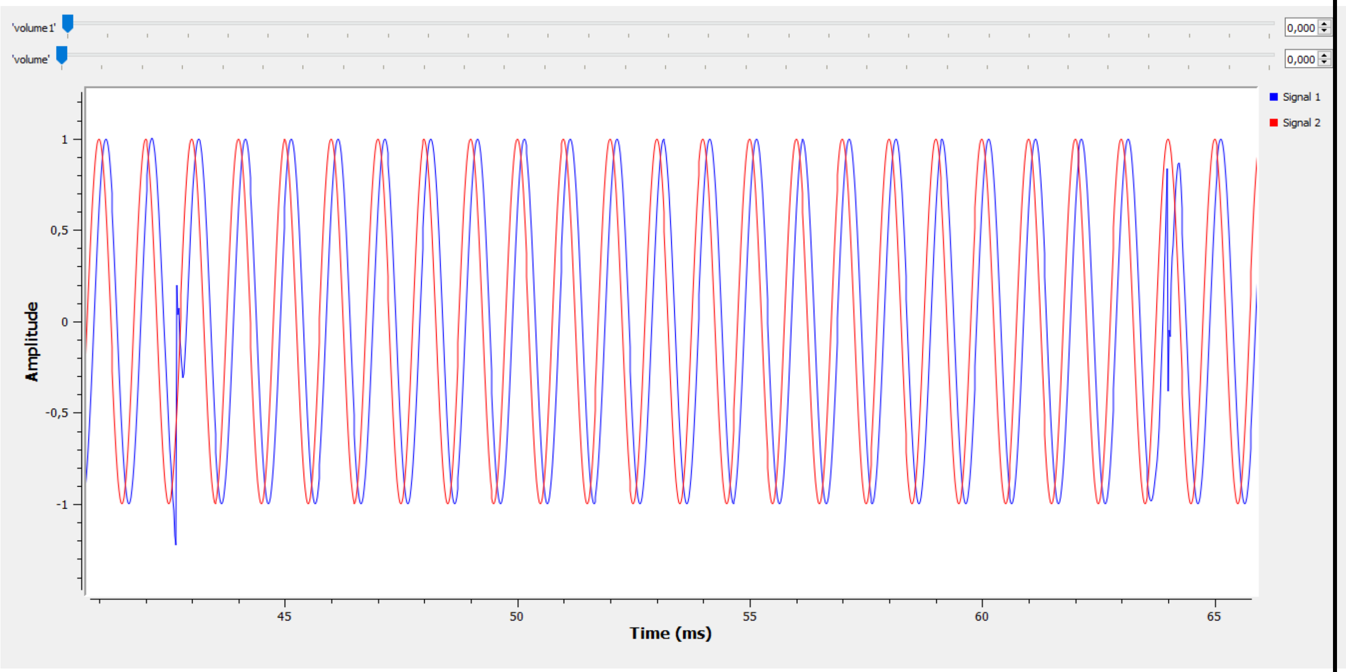


Figure 4.7: Results of the cosine signal simulation using the ZeroPadding method

Based on the visual analysis of the curves in Figure 4.7, we observe a clear improvement in the overall stability of the processed signal (in blue) compared to previous tests without correction:

- Central zone (stabilization): The processed signal (in blue) is almost perfectly superimposed on the original signal (in red). The massive amplitude collapse in the center of the block has completely disappeared.
- Transition zones (block boundaries): However, slight residual transient perturbations are observed, localized around t = 42 ms and t = 64 ms. The addition of zero samples allowed the CQF filters to initialize more smoothly, but the introduction of these artificial values (zeros) still creates a small local break in phase and amplitude.

In conclusion, while the Zero Padding method effectively reduces the destructive impact of edge effects at the ends, it remains limited because it does not guarantee perfect reconstruction and continuity at the junctions of successive blocks. This fully justifies the shift to a more efficient technique: the Overlap method.

4.5. Block MRA analysis of an SDR signal using the Zero-Padding technique

After testing the effectiveness of Zero Padding on a deterministic signal, we apply this technique to a real signal captured by the RTL-SDR hardware interface. The objective is to evaluate the behavior and effectiveness of adding zero samples to a highly noisy, non-stationary, and complex source.

4.5.1. Implementation under GRC

The experimental setup modified under GRC now incorporates our custom analysis block integrating the Padding mechanism, named AMR_CQF_Padding.

As illustrated in Figure 4.8, the demodulated FM signal from the WBFM Receive block (at a sampling rate of 32 kHz) is converted into vectors of 1024 samples. The padding block then inserts null samples to isolate each block of useful data before injecting it into the multiresolution analyzer.

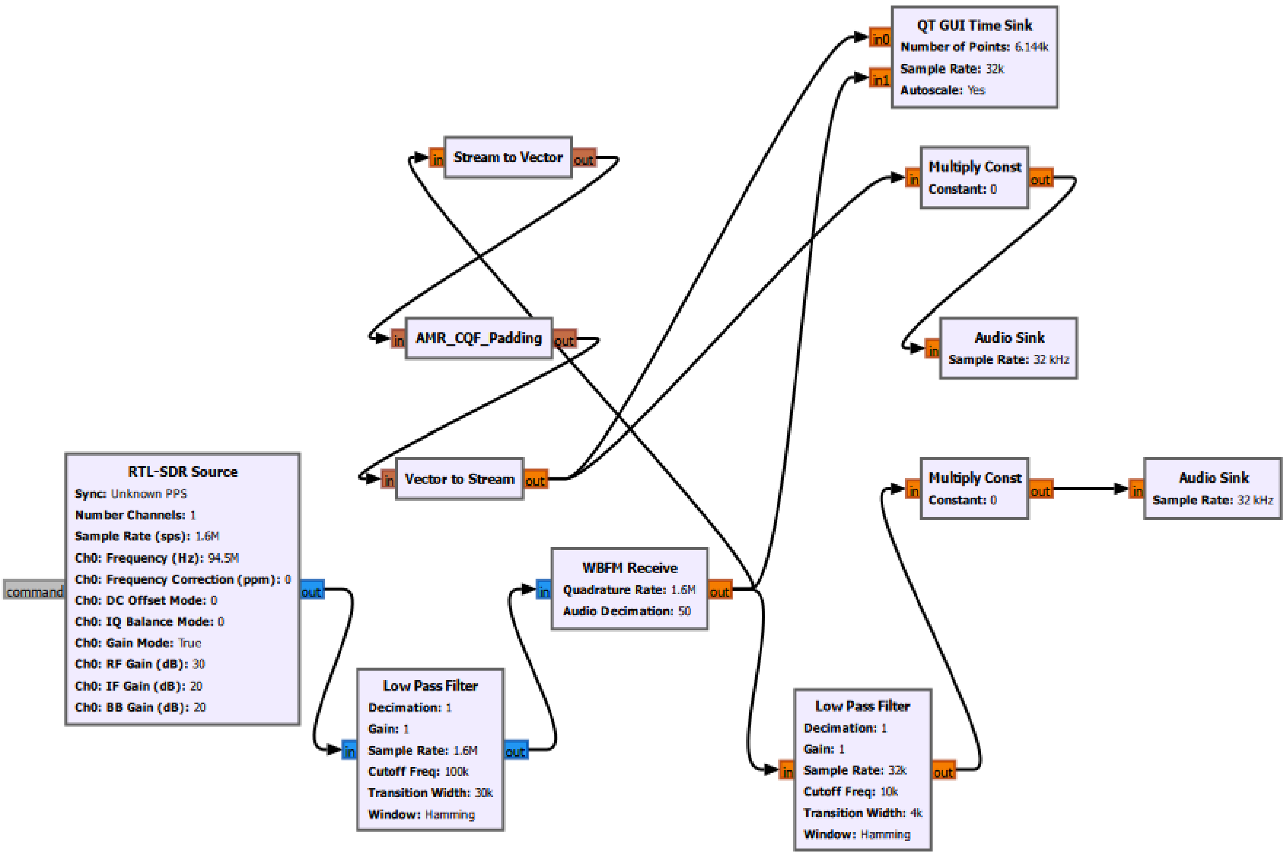


Figure 4.8: Flowgraph of the MRA analysis of an SDR signal using the ZeroPadding method

4.5.2. Results and Analysis

The real-time execution of this flowgraph allows us to obtain the time curves shown in Figure 4.9.

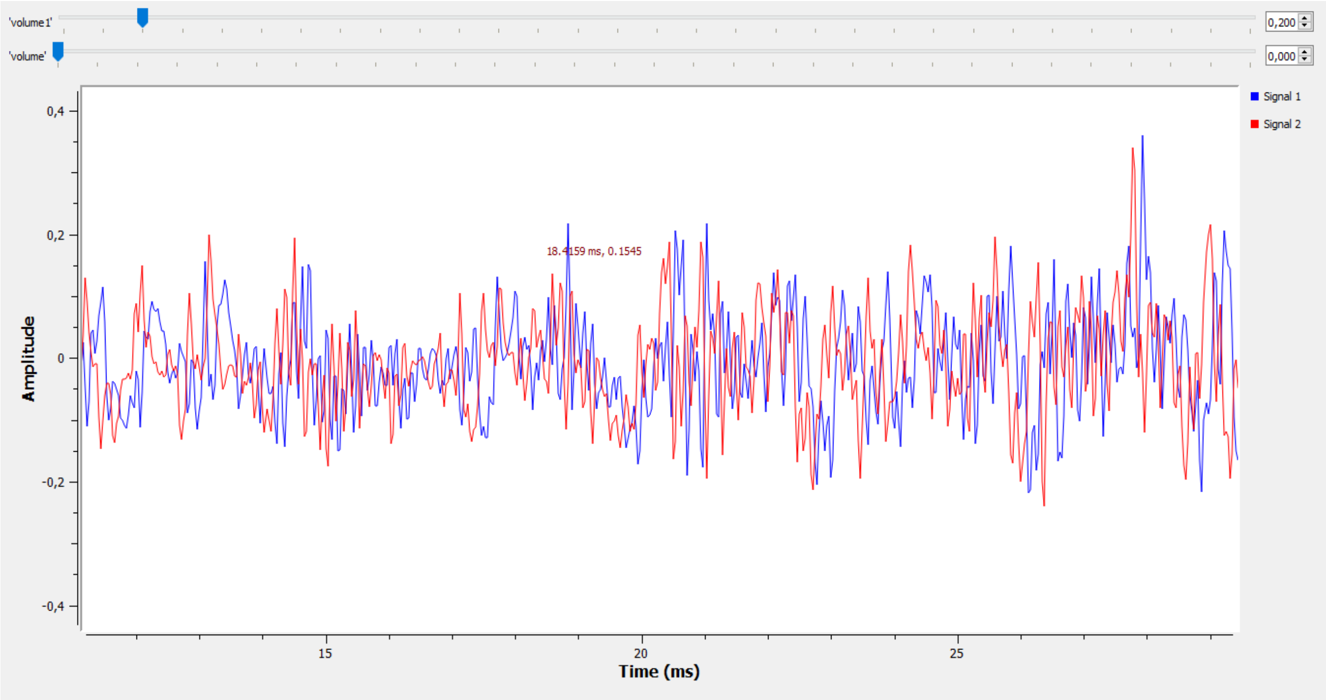


Figure 4.9: Results of the SDR signal simulation using the ZeroPadding method

A comparative visual analysis of the two signals in Figure 4.9 highlights the following technical effects :

Edge effect mitigation: Unlike the results without correction (Figure 4.5), the processed signal (in blue) no longer undergoes a total collapse or a flat plateau at zero. It successfully follows the overall dynamics and variations of the original wave (in red).

Limitations of the method : Although the massive discontinuity has disappeared, discrepancies and slight local amplitude distortions are observed between the blue and red curves (impulsive artifacts). The forced insertion of zero values into a real audio stream creates transient micro-dropouts and local information loss, which degrade the fidelity of the reconstruction

In conclusion, while ZeroPadding offers an improvement over raw block processing, the result remains insufficient for a real-world software-defined radio application due to these residual distortions. This justifies the shift to a more efficient and adaptive approach: the Overlap technique.

4.6. Block MRA analysis of a Cosine signal using the Overlap technique

Although Zero Padding mitigated signal collapse, the introduction of artificial zero values remained a source of residual distortion. To overcome this limitation, we implemented a much more efficient approach: the overlap technique. The goal is to overlap successive blocks to provide the CQF filters with the signal history necessary to correctly calculate the boundaries, thus ensuring perfect temporal continuity.

4.6.1. Implementation of recovery under GRC

In this step, we integrate into the processing chain a custom analysis block developed to manage window overlap, named AMR_CQF_Overlap.

As illustrated in Figure 4.10, the deterministic signal generated by the Signal Source block is divided into vector segments of 1024 samples that partially overlap (depending on the length of the CQF filter). This controlled redundancy masks unstable transient regions of the filters before the final reconstruction of the continuous stream by the Vector to Stream block.

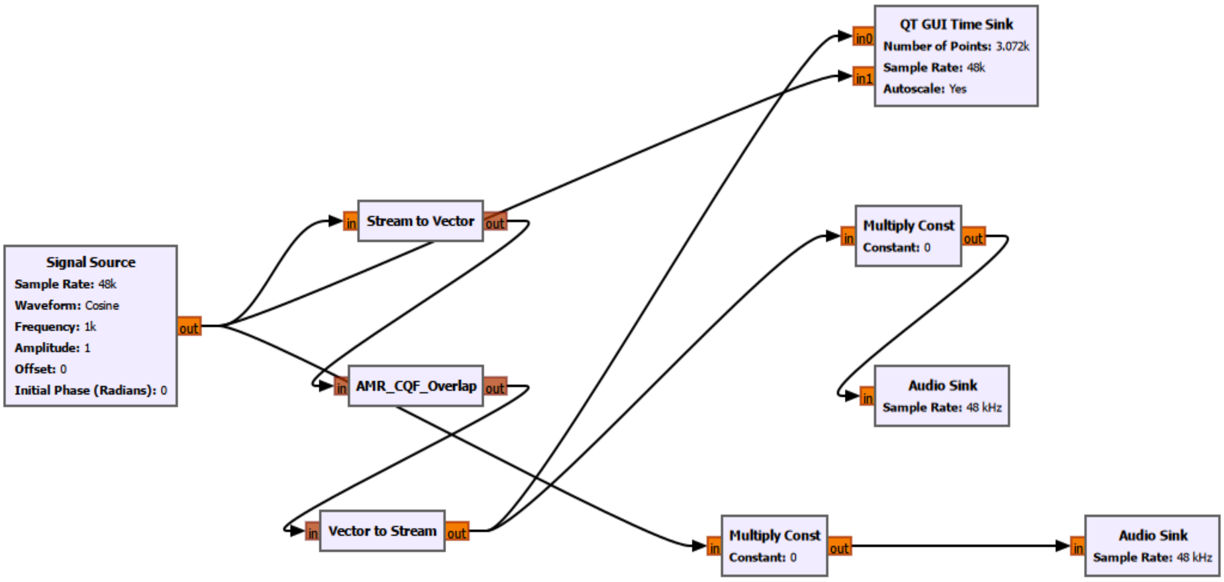


Figure 4.10: GRC flowgraph of the MRA analyzer with the Overlap (Signal Cosine) technique

4.6.2. Analysis of results and validation

The execution of the simulation under GRC allows us to observe on the virtual oscilloscope (QT GUI Time Sink) the time reconstruction shown in Figure 4.11.

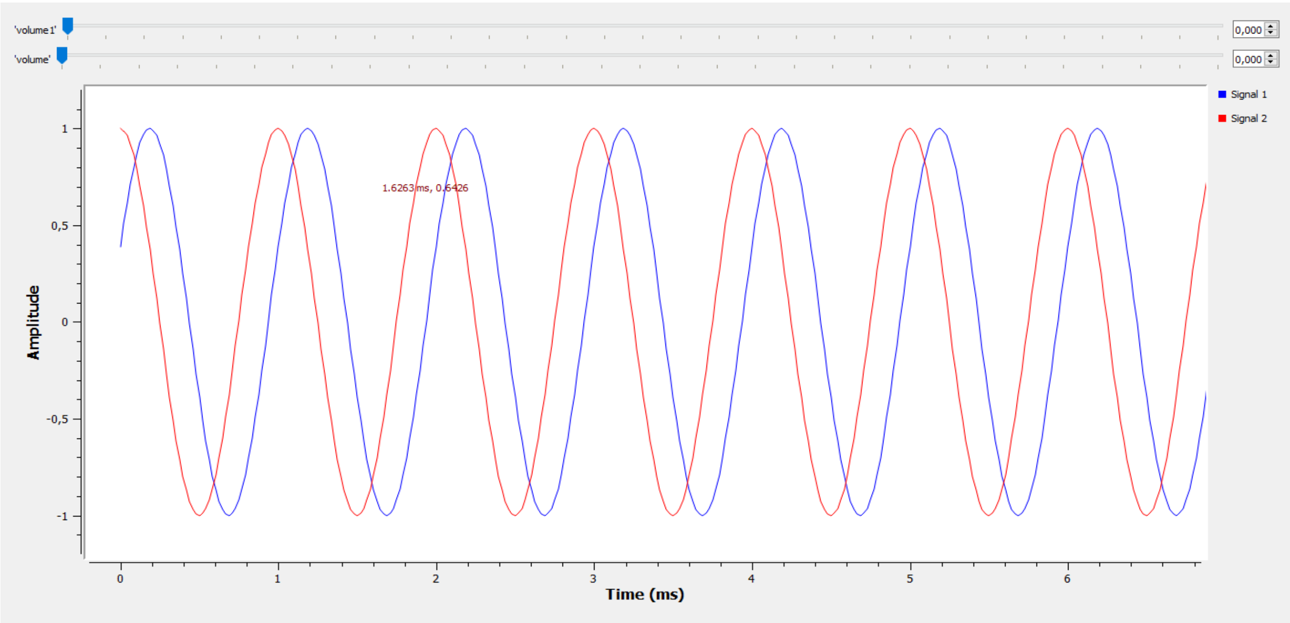


Figure 4.11: Cosine signal perfectly reconstructed after application of the Overlap method

A thorough visual analysis of the signals in Figure 4.11 clearly validates the effectiveness of this method :

Total elimination of the edge effect : Unlike all previous simulations, the processed signal (in blue) no longer shows any sudden jump, amplitude drop, or spurious pulse at the boundaries of the blocks.

Continuity and phase fidelity: The reconstructed waveform is perfectly smooth, stable, and purely sinusoidal over the entire observed time range. The blue signal follows the exact dynamics of the original signal (in red), with a constant and regular phase shift that corresponds to the intrinsic group delay introduced by the digital filtering.

These results confirm that the Overlap technique definitively solves the edge effect problem for a deterministic signal, paving the way for its application to real radio signals.

4.7. Block MRA analysis of SDR signals using the Overlap technique

After validating the effectiveness of Overlap on a deterministic signal (Cosine), we finally apply it to a real, non-stationary, and noisy signal, captured in real time via the RTL-SDR hardware interface. This step constitutes the practical and final validation of our processing block under the constraints of a software-defined radio.

4.7.1. Implementation and Configuration under GRC

The experimental setup implemented under GRC is structured to process the sample stream continuously. As illustrated in Figure 4.12, the RTL-SDR Source block (tuned to 94.5 MHz) captures the complex IQ samples, which are then filtered and demodulated by the WBFM Receive block.

The resulting 32 kHz audio stream is injected into our custom AMR_CQF_Overlap block. This block applies an overlapping sliding window to the 1024 sample blocks, providing temporal redundancy that compensates for the truncation of the CQF filters before the final stream is reconstructed by the Vector to Stream converter.

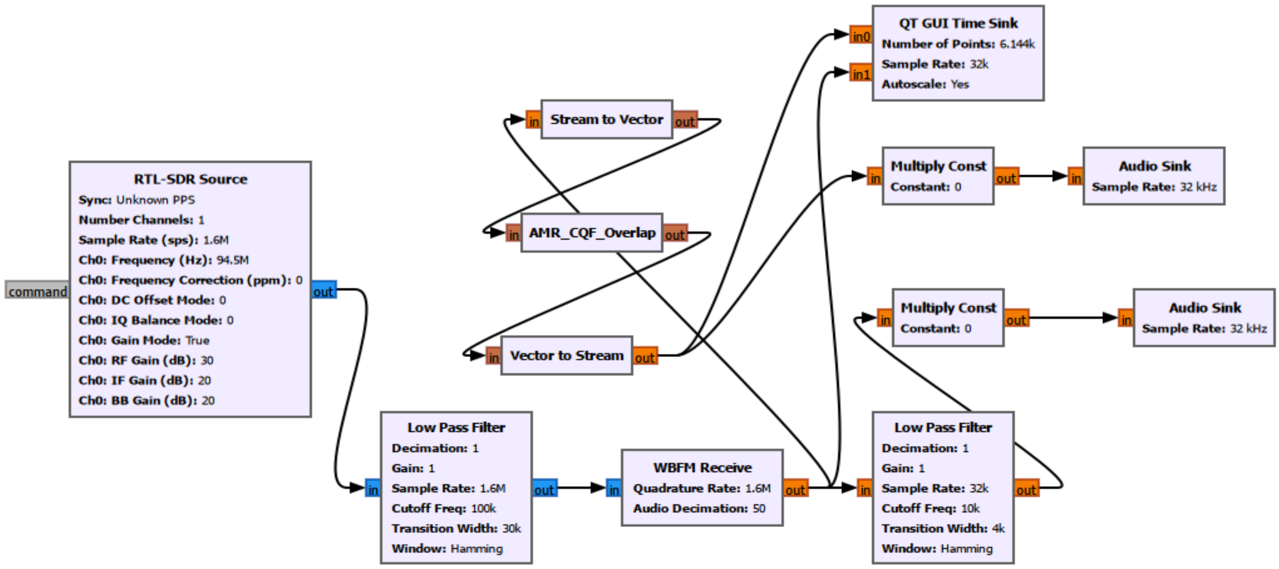


Figure 4.12: Final configuration of the MRA analyzer for overlapping SDR signals under GRC

4.7.2. Results and Interpretation

The execution of this final simulation allows us to analyze the temporal behavior of the reconstructed signals on the virtual oscilloscope (QT GUI Time Sink), shown in Figure 4.13

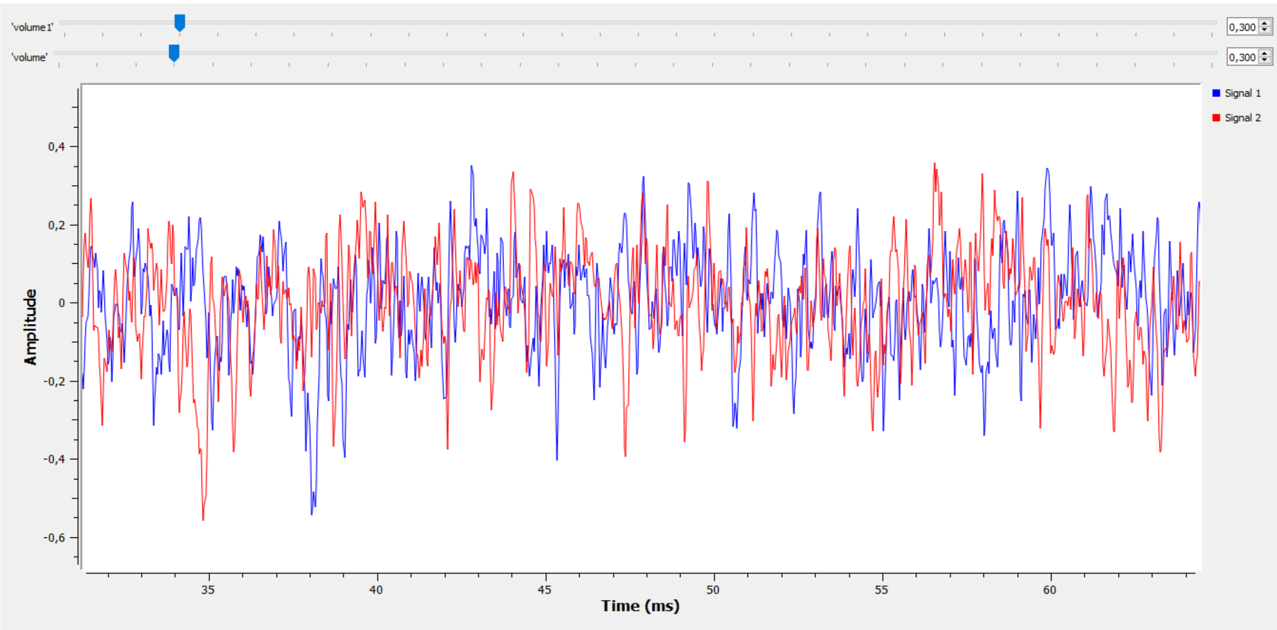


Figure 4.13: Time-domain visualization of the processed SDR signal after application of the Overlap method

The comparative analysis of the curves in Figure 4.13 confirms the robustness and superiority of this approach :

Perfect dynamic tracking: The processed signal (in blue) perfectly overlays and adjusts to the fluctuating and random dynamics of the original audio signal (in red). The curves evolve synchronously across the entire observed time range.

Absolute continuity of the real stream: Unlike tests without correction (Figure 4.5) where the signal collapsed, and Zero-padding (Figure 4.9) which introduced micro-discontinuities, the transitions between successive blocks are completely invisible and smooth. The reconstructed signal after analysis (in blue) shows no phase breaks or artificial amplitude distortion.

These experimental results rigorously demonstrate that the Overlap technique definitively solves the edge effect problem, preserving information integrity and temporal continuity, even on a real, random, and noisy software-defined radio source.

4.8. Block MRA analysis of SDR signals before decimation

To further our investigation and test the performance limits of our algorithm, we propose in this section to modify the placement of the multiresolution analyzer in the software-defined radio receiver chain. Instead of processing the demodulated audio signal at a low sampling frequency (32 kHz), we insert the AMR block directly onto the raw IQ data stream at a very high rate (1.6 MSps), upstream of the WBFM Receive demodulation and decimation block.

4.8.1. Architecture and Implementation under GRC

This configuration requires rigorous adaptation of the flowgraph due to the complex nature of software-defined radio signals. As illustrated in Figure 4.14, the signal from the RTL-SDR Source block first undergoes low-pass filtering.

Since our analyzer processes physical signals, the complex IQ stream is separated into its in-phase (I) and quadrature (Q) components via the Complex To Float block. Both channels are converted into vectors of 1024 samples and then processed in parallel by the

AMR_CQF_Overlap blocks. After the streams are reconstructed by the Vector to Stream blocks, the processed components are reassembled by the Float To Complex block before being fed into the WBFM Receive demodulation stage.

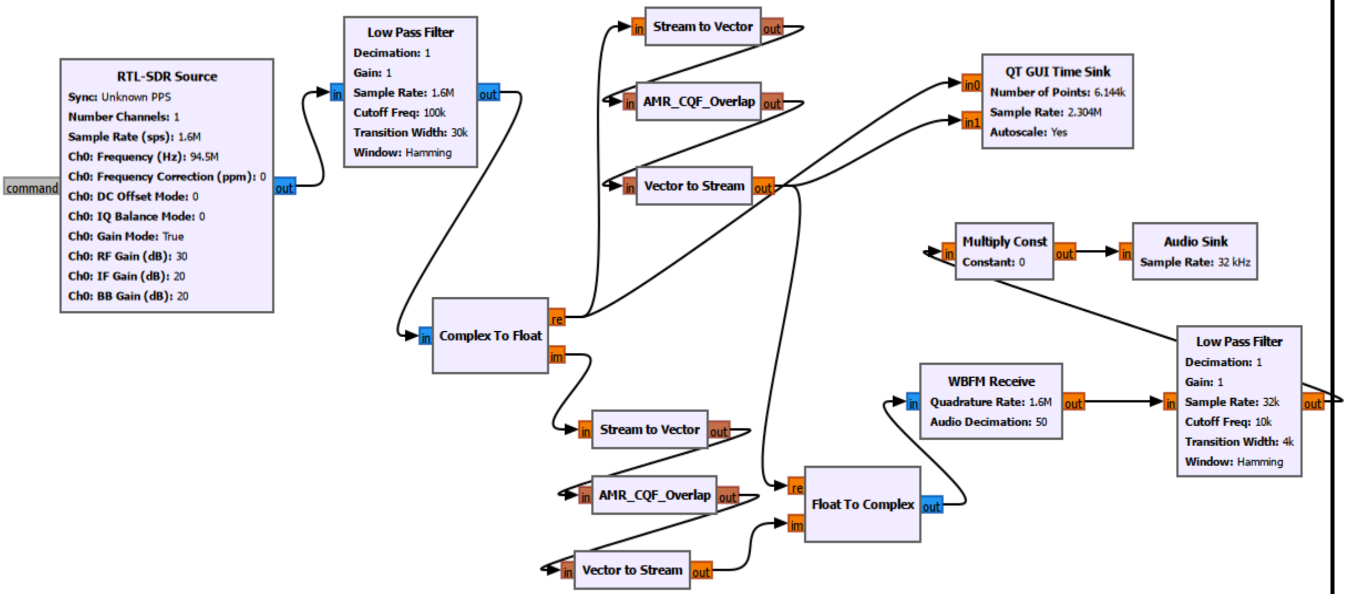


Figure 4.14: GRC flowgraph of the AMR analyzer inserted upstream of the decimation (IQ signals)

4.8.2. Results and Graphical Analysis

The execution of this architecture allows us to observe the signals on the virtual oscilloscope (QT GUI Time Sink), shown in Figure 4.15.

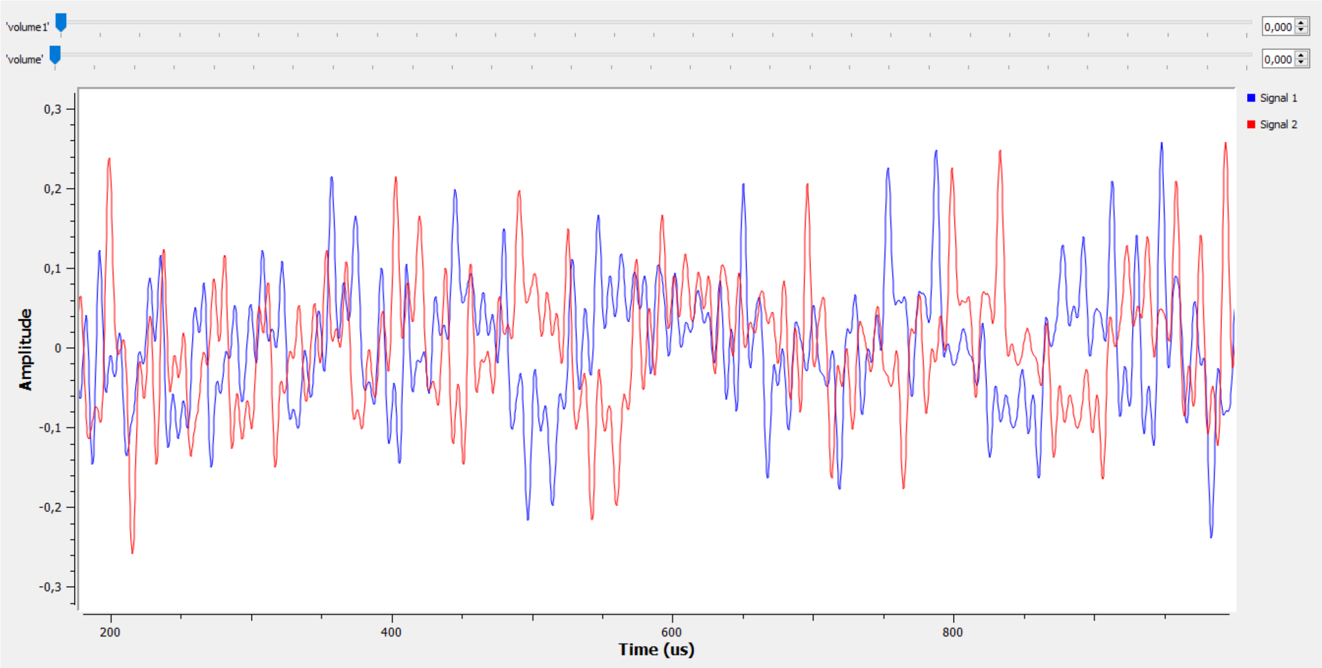


Figure 4.15: Simulated SDR signals at high sampling frequency (scale in ms)

Analysis of Figure 4.15 highlights fundamental technical aspects:

- Time scale change: The time axis scale is now expressed in microseconds (ms), which visually validates the fact that the processing takes place at a high sampling rate (1.6 MSps), well before audio decimation.
- Fidelity and robustness of IQ processing: The processed signal (in blue) is superimposed with extreme precision on the reference signal (in red). Despite the speed of the flow and the complexity of the waveform, the Overlap technique completely eliminates edge effects. Fast transitions and high-frequency peaks are faithfully reconstructed without any distortion or phase divergence.

This experiment successfully demonstrates that our implementation of multiresolution block analysis with overlap is highly stable and capable of supporting real-time, high-bandwidth data streams.

4.9. Optimization proposal: Continuous Multiresolution Analysis

Although the Overlap method solves the edge effect problem induced by vector segmentation, it imposes redundant calculations and buffer management, which increase the overall system latency. To overcome these constraints, this section proposes an optimized alternative that we call Continuous Multiresolution Analysis. The fundamental idea is to completely eliminate the notion of blocks by applying filter banks directly to the continuous sample stream, sample by sample.

4.9.1. Validation on a Deterministic Signal (Cosine)

This approach relies on the exploitation of GNU Radio's native filtering structures. As illustrated in Figure 4.16, the reference sinusoidal signal (1 kHz at a rate of 32 kHz) undergoes tree decomposition without any prior vector conversion.

The two-level MRA analysis and synthesis are implemented in a cascade using the Decimating FIR Filter block (for calculating the approximation and detail coefficients with decimation by 2) and the Interpolating FIR Filter block (for the reconstruction stage with interpolation by 2). The coefficients of the associated filters (H0, H1, G0, G1) correspond to the theoretically calculated CQF orthogonal filters.

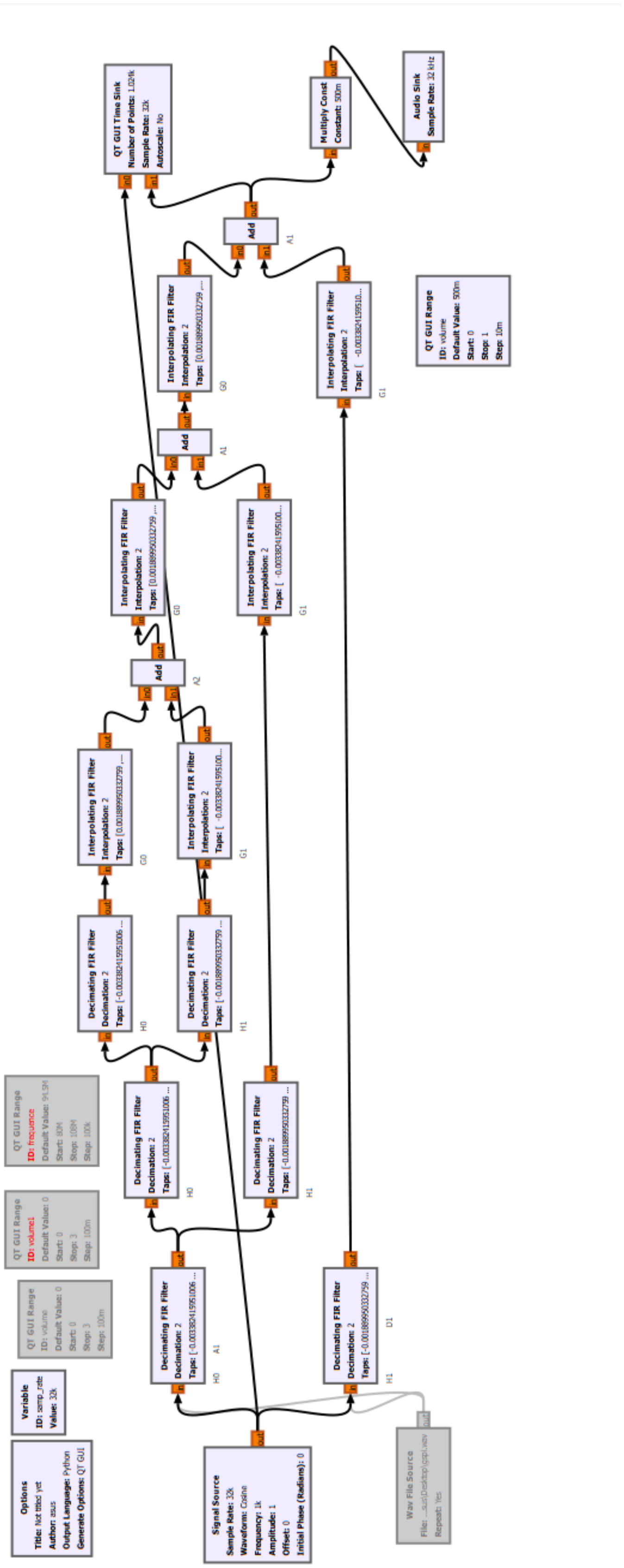


Figure 4.16: GRC flowgraph of the AMR analyzer in continuous mode for a cosine signal

The temporal observation of the reconstructed signal via the graphics sink (QT GUI Time Sink) is shown in Figure 4.17.

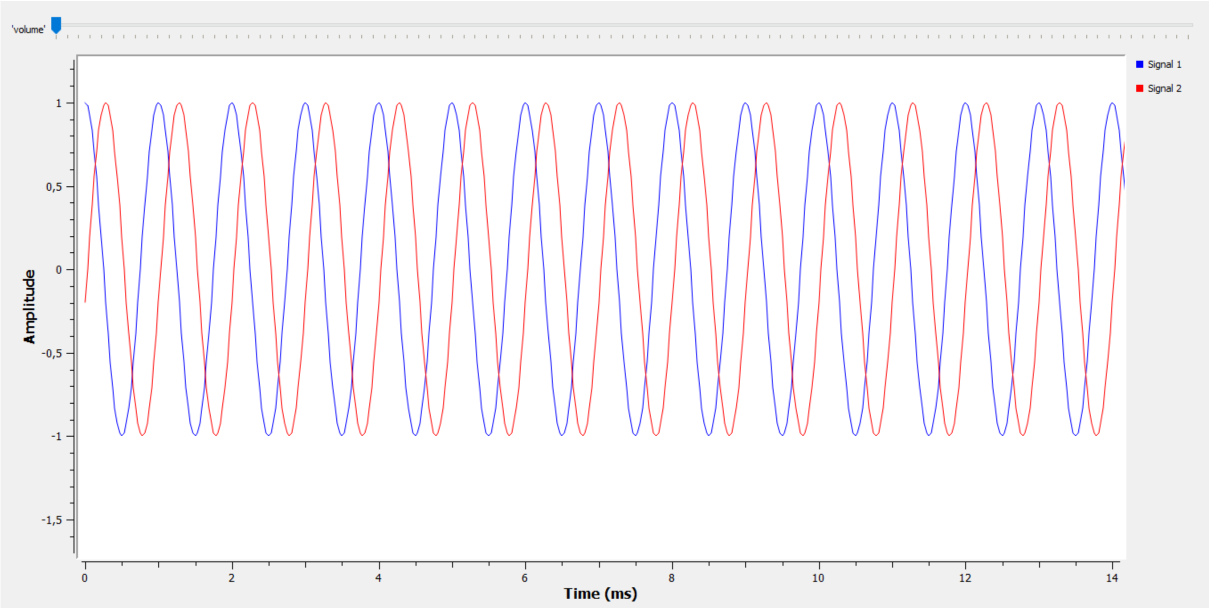


Figure 4.17: Reconstruction of a cosine signal by continuous AMR (Stream-Based)

Analysis of Figure 4.17 shows perfect continuity. The absence of block segmentation natively and definitively eliminates any possibility of edge effects. The processed signal (in blue) is perfectly stable from the first sample and faithfully follows the source signal (in red).

4.9.2. Final Application on Continuous SDR Physical Signal

To definitively validate this optimization proposal, we apply this continuous architecture onto a real stream captured by the RTL-SDR hardware interface.

As illustrated in Figure 4.18, the audio data stream demodulated by the WBFM Receive block (clocked at 32 kHz) undergoes the same continuous tree-like processing by FIR (Analysis/Synthesis) filters sample by sample, thus eliminating the Stream to Vector and Vector to Stream conversion blocks.

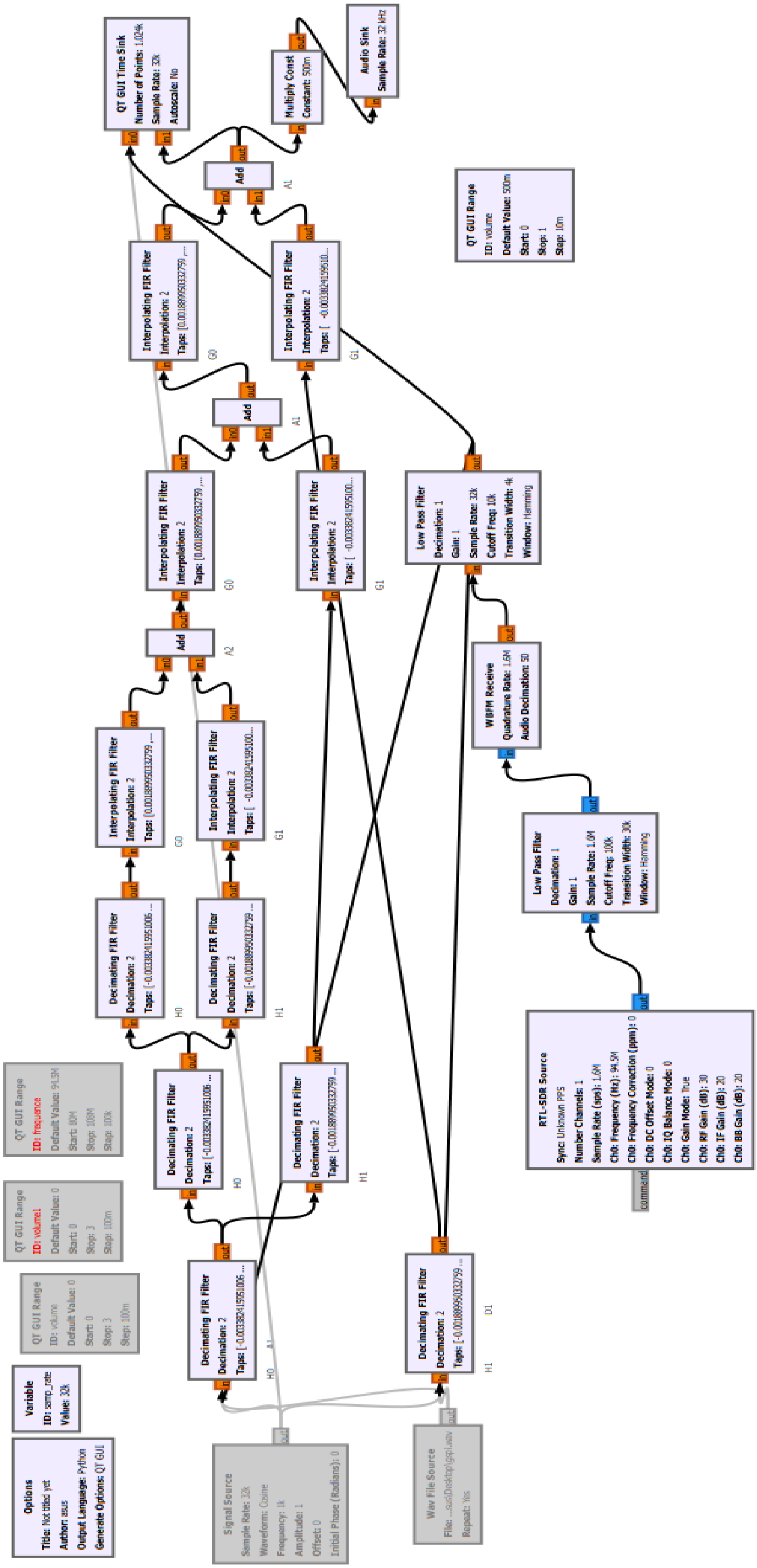


Figure 4.18: Final configuration of the MRA analyzer in continuous (Stream-Based) mode for SDR signals

4.9.3. Results and Graphical Analysis

The extraction and visualization of real-time signals via the QT GUI Time Sink are shown in Figure 4.19.

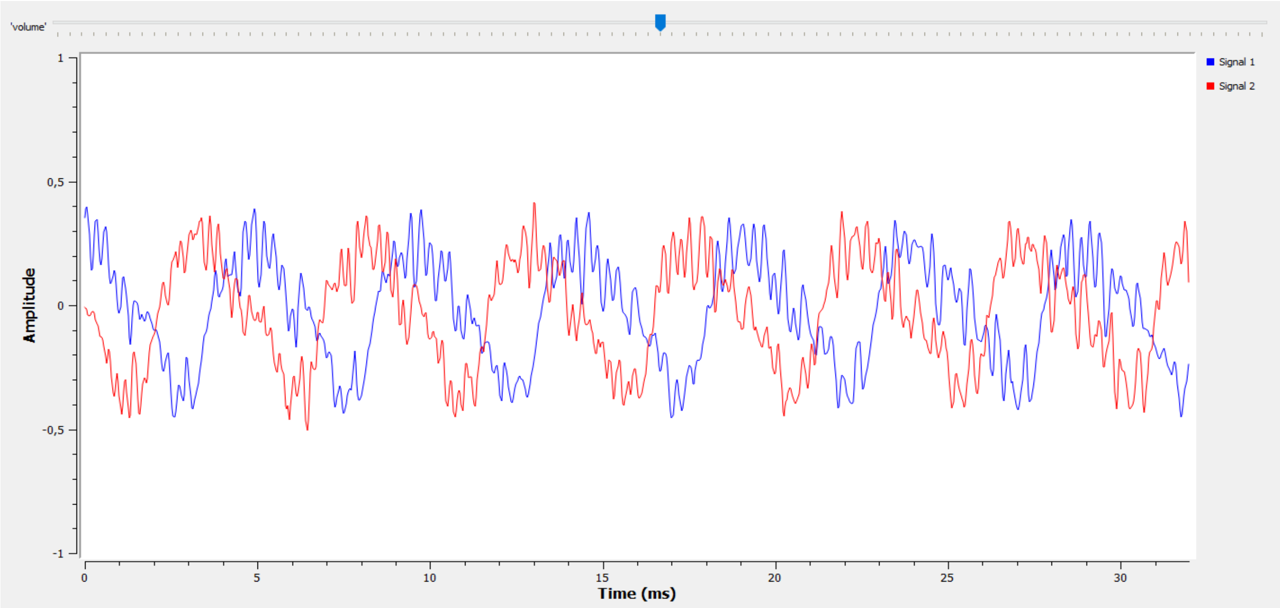


Figure 4.19: Physical SDR signals processed by mra in continuous (Stream-Based) mode

Examination of Figure 4.19 confirms the effectiveness of the continuous approach on a real and dynamic signal:

- Absolute suppression of discontinuities: Unlike raw block tests, the reconstructed signal (in blue) shows no amplitude jumps, no micro-discontinuities, and no transient artifacts over the entire time scale.
- High-fidelity dynamic tracking: Although sampling is done at a low frequency (32 kHz) after demodulation, the blue curve perfectly follows the complex morphology, rapid fluctuations and intrinsic noise of the original radio signal (in red).
- Constant group delay : We observe a constant and uniform time shift between the two signals, which corresponds strictly to the delay induced by the impulse response of the FIR filters of the tree, without any dynamic phase drift.

This real-time implementation demonstrates excellent stability and a drastic reduction in processing latency. By operating in stream mode, the algorithm overcomes windowing constraints while maintaining complete reconstruction fidelity in the face of variations from a real radio source.

Note : It is important to emphasize that the developed Multiresolution Analysis (MRA) operates optimally for demodulated SDR signals, that is, when the sampling frequency is low (32 kHz audio sampling rate). This technical success validates the viability of our approach and leads us to propose, in what follows, new perspectives for extending its performance.

4.10. Perspectives and Optimization Axes

Although our study successfully validated the application of continuous and block Multiresolution Analysis (MRA) (with overlap) to demodulated SDR signals at low sampling rates, several promising avenues of research are opening up to optimize and extend the performance of our system in real time. We propose two major areas of development:

4.10.1. Wavelet denoising of SDR signals using MRA

The first approach involves using MRA not only for time-domain analysis but also as an active denoising tool through wavelet thresholding. Since signals captured by RTL-SDR equipment are heavily affected by thermal noise and radio frequency interference, applying a thresholding algorithm (soft or hard thresholding) to the detail coefficients derived from the MRA would allow for the isolation and elimination of noise before reconstruction. The goal is to significantly improve the signal-to-noise ratio (SNR) and the final quality of the demodulated signal by leveraging the optimal time-frequency localization of the wavelets.

4.10.2. Polyphase implementation of MRA before demodulation (High Sampling Frequency)

The second proposal aims to overcome the computational complexity barrier encountered when processing raw IQ signals at very high sampling rates (1.6 MSps) prior to demodulation. We propose implementing a polyphase structure for the MRA CQF filter bank. This polyphase decomposition allows the downsampling step to be switched to a position prior to the actual filtering. By performing digital filtering at a reduced frequency, this technique drastically reduces the number of operations per second (multiplications and additions). This enables the

MRA to be integrated directly into the broadband RF stream before any analog-to-digital conversion, while maintaining minimal algorithmic complexity and smooth real-time execution.

Conclusion

This fourth chapter was entirely dedicated to the practical implementation, evaluation and optimization of a Multiresolution Analysis (MRA) system under the GNU Radio environment, by comparing theoretical models with the constraints of real signals from an RTL-SDR interface.

The steps in our scientific approach have allowed us to draw the following fundamental conclusions:

1. The problem of the edge effect : Raw block processing (isolated windows of 1024 samples) generates critical distortions and massive signal breaks at the ends of the windows, whether it is a simulated deterministic signal or a real demodulated FM signal.
2. The inadequacy of Zero-Padding : Inserting null samples at the periphery of the blocks helps to dampen the amplitude jump but proves insufficient for real-world applications. It generates micro-dropouts incompatible with a high-fidelity communication stream.
3. The robustness of the Overlap technique : The overlap technique proved to be the ideal solution for block processing. By ensuring temporal redundancy, it eliminates edge effects and guarantees perfect signal continuity.
4. The advantage of continuous (Stream-Based) mode:Finally, our proposal for a continuous MRA architecture has demonstrated that by completely eliminating vectorization, we natively remove the source of the edge effect problem while optimizing overall latency, thus offering a very high-performance alternative for industrial software-defined radio applications.

All of these experimental validations confirm the viability and robustness of our customized software-defined radio platform, thus opening up solid prospects for the deployment of advanced real-time signal processing algorithms.

GENERAL CONCLUSION

This dissertation was situated at the convergence of two major pillars of telecommunication : the architectural flexibility of Software-Defined Radio (SDR) and the analytical power of Wavelet Multiresolution Analysis (MRA). The central ambition of this research was to overcome a persistent obstacle : The degradation of spectral and temporal fidelity induced by edge effects during the real-time processing of continuous data streams. By combining mathematical formalization, algorithmic modeling, and empirical validation on physical radio signals, this study provided rigorous solutions to the limitations of standard computational approaches. On a theoretical level, the investigation conducted in the first chapters highlighted the elegance and necessity of complex envelope representation (I/Q signals) to overcome the rigidities of traditional analog chains. However, the application of SDR architectures to non-stationary signals has highlighted the epistemological limitations of classical Fourier analysis. Its inability to adapt its tiling of the time-frequency plane, has justified the introduction of the wavelet transform as a localized and adaptive analysis operator. The core of our problem crystallized during the transposition of Mallat's hierarchical decomposition algorithm to the finite nature of digital signals. The truncation inherent in block processing generates discontinuities at the ends of data vectors, creating severe digital artifacts within conjugate quadrature (CQF) filter banks. This phenomenon, far from being mere residual noise, substantially alters the approximation and detail coefficients, thus compromising the condition for perfect signal reconstruction. The major and original contribution of this thesis lies in the integration of these concepts within the open-source development environment GNU Radio, materialized by the design and evaluation of custom processing blocks. The experimental approach, conducted on simulated deterministic signals as well as on complex signals captured in real time (FM broadcast band), allowed for the formulation of major scientific conclusions: Error characterization: The destructive impact of edge effects was mathematically quantified, revealing the propagation of distortions through the decomposition levels. Validation of discrete fixes: Classical boundary regularization methods (notably Zero-Padding and Overlap techniques) were implemented and evaluated, demonstrating a significant restoration of signal fidelity at the cost of a quantifiable compromise in computational load and algorithmic latency. Algorithmic paradigm shift: To overcome the limitations of block processing, an innovative Continuous Multiresolution (Stream-Based) Analysis architecture was developed. By adapting filtering structures to treat the signal as an uninterrupted stream, this methodology natively eliminates the recurrence of edge discontinuities and minimizes overall latency, thus offering a high-performance solution for real-time signal processing applications. In short, this work demonstrates that the industrial application of the mathematical concepts of multiresolution analysis cannot bypass rigorous optimization of software data structures. The synergy developed in this dissertation between formal rigor and software engineering lays a solid foundation for the deployment of wavelet algorithms within future telecommunications standards.

References:

[1] J. Mitola, “Software Radio: Survey, Critical Analysis and Future Directions,” IEEE Aerospace and Electronic Systems Magazine, vol. 8, no. 4, pp. 23-36, April 1993.

[2] D. Wilson, "Emerging technology: Will software-defined radio shake up communications", Electronics Design, Strategy, News, vol. 2009, Aug 2007.

[3] GI Wassi, "Radiolocation in a non-stationary confined environment", Master's thesis, LAVAL University Quebec, Canada, May 2005.

[4] JB. Dutertre, "Design of integrated circuits: A/D Converters", Course at the École des Mines de Saint-Étienne.

[5] W. Tuttlebee, "Software Defined Radio: Baseband Technologies for Transceivers", John Wiley & Sons, 2004.

[6] ENSIMAG, "Digital Techniques for Transmissions: Digital Modulations", University of Grenoble, 2008.

[7] B. Razavi, “Principles of Data Conversion System Design”, Wiley-IEEE Press, 1995.

[8] C. Prévot, "Analog-to-Digital Converters", Techniques de l'Ingénieur, Treatise on Electronics.

[9] Great Scott Gadgets, “HackRF One: Software Defined Radio Peripheral,” Hardware Engineering Specifications.

[10] Analog Devices, “High Speed Data Converters: Architectures and Applications,” Technical White Paper.

[11] Analog Devices, "ADALM-PLUTO: Software Defined Radio Learning Module", Technical Specifications and System-on-Chip (SoC) Architecture Report, 2017.

[12] Realtek Semiconductor Corp., “RTL2832U: High-Performance Network Interface Controller and Demodulator,” Technical Datasheet and Specifications Report.

[13] Xilinx Inc., "Zynq UltraScale+ RFSoC Data Sheet: Overview", DS889 (v1.5), 2020.

[14] Texas Instruments, “Understanding ADC Architectures: Pipeline ADCs,” Application Report, 2022.

[15] Analog Devices, "AD9446: 16-Bit, 80 MSPS/100 MSPS ADC Data Sheet", Rev. C, 2021.

[16] S. Mallat, A Wavelet Tour of Signal Processing: The Sparse Way, 3rd ed., Academic Press, 2008.

[17] F. Hlawatsch and F. Auger, Time-Frequency Analysis, Wiley-ISTE, 2005.7

[18] C. Charles, "From the Fourier transform to the wavelet transform", Interstices, 2012. Available at:interstices.info.

[19] J. Gilles, Introduction to Wavelets, Course Notes, SDSU, 2021.
[20] M. Pan, "Discrete Wavelet Transform Vs Short-Time Fourier Transform", Technical Report, ResearchGate, 2017.
[21] I. Daubechies, Ten Lectures on Wavelets, SIAM, 1992.
[22] S. Jaffard, Y. Meyer, and R. Ryan, Wavelets: Tools for Science & Technology, SIAM, 2001.
[23] MathWorks, Wavelet Toolbox™ User's Guide, R2022b.
[24] J. Morlet et al., "Wave Propagation and Sampling Theory," Geophysics, 1982.
[25] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," Math. Ann., 1910.
[26] R. Polikar, The Wavelet Tutorial, 2nd ed., 2001.
[27] Y. Meyer, Wavelets and Operators, Hermann, 1990.
[28] G. Strang and T. Nguyen, Wavelets and Filter Banks, 1996.
[29] M. Vetterli and J. Kovačević, Wavelets and Subband Coding, 1995.
[30] CS Burrus et al., Introduction to Wavelets and Wavelet Transforms, 1997.
[31] RE Crochiere and LR Rabiner, Multirate Digital Signal Processing, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[32] MathWorks, "Practical Introduction to Multiresolution Analysis," Signal Processing Toolbox Documentation, R2023b. Available online:www.mathworks.com.
[33] M. Bellanger, Digital Signal Processing: Theory and Applications, 9th edition, Dunod, Paris, 2012.

[34] A. Cohen, Wavelets and multiresolution, Éditions de l'École Polytechnique, Masson, Paris, 1992.

[35] I. Daubechies, Ten Lectures on Wavelets, SIAM, 1992.

[36] T.-X. He, Wavelet Analysis and Multiresolution Methods, CRC Press, 2000.

[37] T. Barth and T. Chan, Multiscale and Multiresolution Methods, Springer, 2002.

[38] RT Yazıcılıgil et al., Software Defined Radio using MATLAB & Simulink, MathWorks, 2018.

[39] MathWorks, "Dealing with Border Distortion," Wavelet Toolbox Documentation, R2023b. Available online:https://www.mathworks.com/help/wavelet/ug/dealing-with-border-distortion.html