

Dynamic Energy Quality Optimization through Reversible Adaptive Filtering in Reconfigurable Datapath Architectures

Arun Raj S.R¹, Dr. G. Ramana Murthy²

Abstract

Reconfigurable adaptive filter is a revolutionary reconfigurable data route architecture that is presented in this work with the purpose of resolving the ever-changing trade-off between energy quality and adaptive filtering that takes place. The four adaptive filtering techniques that reconfigurable adaptive selects dynamically are Least Mean Squares (LMS), Partial Update Normalized LMS (PU-NLMS), Set-Membership Normalized LMS (SM-NLMS), and Normalized LMS (NLMS). This selection is based on the grading difficulty levels that are present during runtime. The design places an emphasis on reusing modules, which results in a compact implementation of VLSI hardware that functions via the use of reversible logic techniques. As part of this study, we came up with a plan for an 8x8 multiplier circuit that uses a Feynman gate (FG) full adder along with a Press Gate (PG) design flow and can be turned around. This design resulted in a reduction in the depth of the multiplication module within the framework of a reconfigurable adaptive filter architecture. Synthesis assessments conducted by Xilinx have shown that this reversible design results in superior performance when compared to traditional architecture. There are several parts to a case study that looks at reconfigurable adaptive algorithms. These include an accurate resorting divider, a 5G input sequence with noise and the desired signal, and a link to the update control block of the LMS adaptive filter. The outcomes of synthesis research that compared two distinct multiplier designs on a Vertex-5 FPGA with four distinct filter algorithm levels gave information on power, latency, and area. In this study, it is shown how to use reconfigurable adaptive filters to get good adaptive filtering with changing energy quality using adaptive filters.

Keywords: LMS, PU-NLMS, SM-NLMS, NLMS, Wallace Tree Multiplier, Feynman gate, Press gate.

I. Introduction

In signal processing applications, adaptive filtering is an extremely important component since it provides the power to dynamically alter filtering methods in response to different input circumstances. To solve the problem of quality being an explicit design requirement, energy-quality scalable (EQ- scalable) methods strike a balance between energy consumption and quality at different levels of computing abstraction. Many error-tolerant applications, including AI, digital signal processing, image computing, and video processing, make use of scalable equalization schemes that trade energy for quality loss. The EQ trade-off that is inherent in adaptive filtering is a substantial challenge,

¹ Research Scholar, Department of ECE, ACED, Alliance University, Bengaluru, India

² Professor, Department of ECE, ACED, Alliance University, Bengaluru, India

necessitating the development of creative methods in order to strike a balance between computational complexity and performance. In order to solve this difficulty, our research presents a reconfigurable adaptive filter that is based on a datapath architecture. The ever-changing energy-quality trade-off in real-time applications inspired the creation of this design. One of the most important tools for researching EQ-scalable VLSI systems in real time is adaptive filtering (AF) [1]. Audio and speech signal manipulation, communication over wireless networks, and medical signal processing are just a few of the many digital signal processing (DSP) domains that make use of AFs, which are well-established filtering methods.

AF methods are crucial in scenarios where the input signals' characteristics change over time. On the other hand, the intrinsic trade-off that exists between the amount of computing energy used and the quality of filtering presents a hurdle. It may be difficult for traditional fixed designs to adjust adequately to dynamic situations, which may result in performance that is less than ideal and an increase in energy consumption [2]. The need for a flexible and dynamic solution that is capable of navigating the energy quality trade-off in AF in a smooth manner is what prompted the development of the reconfigurable adaptive filter architecture. The reconfigurable adaptive filter is designed to dynamically pick among a range of AF methods, with the goal of enhancing performance in response to changing needs. The introduction of a reconfigurable datapath and the use of data gating techniques enable this.

The requirements of the application for filtering accuracy are highly dependent on the intensity of the signal and the noise environment and are not set throughout runtime. Depending on the runtime environment and the magnitude of the signal, each AF technique may provide a unique set of benefits. In high-level noise scenarios, AFs with greater complexity are often the best choice. Worst-case energy use derives from the system designers' decision-making process based on worst-case noise scenarios [3]. Switching to a more complicated AF to lead in a low-noise environment, on the other hand, is a power-consuming decision. Furthermore, in a situation with low noise, picking an AF that is less complicated results in a less inaccuracy due to the inherent filtering noise, but getting an AF that is more complex results in a higher cost per unit of time. It is possible that considerably increasing energy efficiency may be accomplished by scaling the AF algorithm of VLSI designs in a dynamic way. Developing a flexible data processing unit that is capable of dynamically selecting algorithms is the primary objective of this study, which aims to overcome the challenge of creating such a unit. The goal is to maximize energy consumption while also managing and limiting noise during runtime. Using shared common blocks can help achieve adaptive filtering blocks, which have similarities. This work introduces reconfigurable AF, a VLSI design for an EQ-scalable adaptive filter with a customizable datapath. The reconfigurable AF design concept shows how to effectively use a flexible architecture that can handle four different adaptive algorithms by making good use of the parts that they share [4].

In order to provide real-time EQ scalability, reconfigurable AF utilizes the data-gating approach to selectively activate or disable blocks based on the specific needs of each filtering architecture. By changing the datapath operation flow, the customizable AF structure allows the user to change the algorithms' complexity while the system is running. The LMS, NLMS, PU-NLMS, and SM-NLMS are the four distinct algorithms that may be selectively implemented as a result of this. In order to offer a condensed hardware implementation, each of the reconfigurable AF architecture's LMS-based filters shares modules [5]. When the data from the dormant modules is gated, the reconfigurable AF is able to achieve energy savings. This is accomplished by eliminating the switching activity of the modules, which in turn reduces the amount of dynamic power consumption across all of the various operating modes. The reconfigurable AF operation uses two distinct coefficients in the context of a case study that focuses on reducing interference in electroencephalogram (EEG) data.

The fundamental goal of this study is to create a reconfigurable adaptive filter that not only adjusts to changing data processing needs but also optimizes hardware resources. The four alternative datapath topologies provide flexibility in managing a wide range of data types and processing situations. However, flexibility comes at the expense of larger logic sizes, longer critical route delays, and greater power consumption, demanding a creative solution. The heart of the proposed structure is the use of reversible logic approaches, notably a reversible Wallace tree multiplier. This multiplier design makes use of the Feynman Gate (FG), Toffoli Gate (TG), and Peres Gate (PG), with the goal of reducing logic size, critical path delays, and power consumption [6]. The reversible nature of these logic units offers promise for delivering efficient and dynamic signal processing capabilities in a small physical footprint. Because adaptive filters play an important role in real-time applications, high-speed processes become critical. The inclusion of the reversible Wallace tree multiplier not only overcomes the aforementioned difficulties but also offers faster processing rates, making the proposed architecture ideal for applications that need real-time adaptation and responsiveness. This introduction lays the groundwork for a detailed examination of the proposed reconfigurable adaptive filter architecture. The next parts will go into the complexities of the four datapath designs, the use of reversible logic, and the benefits of the reversible Wallace tree multiplier. This study will use detailed analysis and testing to determine the efficacy and efficiency of the proposed architecture in addressing the needs of modern signal processing applications.

As of right now, the following are our most recent thoughts for this research: 1) In order to construct a VLSI filtering system that is both energy-efficient and scalable, we make use of a reconfigurable datapath. This datapath has a reversible wallace tree multiplier and changeable complexity at runtime, and it allows us to choose four adaptive filter algorithms. We also compare the standard design of four alternative adaptive filter architectures to our proposed design. 2) We show that the reconfigurable adaptive filter approach can continuously scale in terms of energy quality, logic size, and critical path latency at runtime [7]. Additionally, we contrast a static method with a dynamic one that makes decision adjustments during execution based on the input signal-to-noise ratio (SNR). 3) We test the AF approach in different noise scenarios to find the compromise between power consumption, hardware size, bandwidth, maximum clock frequency range, accuracy, and other parameters for different circuit modes of operation. The following sections of this work are grouped as follows: Section 2 describes the review of related work on the reconfigurable adaptive filter, and Section 3 includes an in-depth discussion and comparison of the proposed reversible Wallace tree multiplier to traditional multiplier designs. Section 4 describes the proposed four distinct reconfigurable AF architectures, including their design ideas and functionality. Section 5 covers the synthesis results and performance indicators. Finally, Section 6 finishes the work with a summary of the results and recommendations for further research.

II. REVIEW OF RELATED WORK TO THE RECONFIGURABLE ADAPTIVE FILTER

Methods for developing accurate and high-performance DSP systems have been detailed in a number of publications throughout the years. One interesting way to get a clean signal is to use an adaptive filter, in this case, the LMS family filter, to remove artifacts from medical data. The study investigated the efficacy of several LMS-based filters in mitigating power line interference in ECG data. These filters included NLMS, SM, and signed families. The authors continue to assess the filters' authority even when they only reveal their algorithmic actions. There is also a comparison of the adaptive LMS filter method with the LPF Butterworth filter and the wavelet function. In order to eliminate artifacts, the analysis was performed on the 5G signal [8]. Surpassing the LPF Butterworth filter, the LMS filter demonstrated superior efficiency with MSE and peak signal-to-noise ratio (PSNR) metrics centered around wavelet values.

These filters are used in a system that removes artifacts from electrocardiogram (ECG) data. To build a circuit using as little power and space as possible, the writers provided simplified designs. The system's filter efficiency was tested as part of the endeavor, which included building and synthesizing the ASIC designs utilizing ST 65nm technology. Indicators of quality, including RMSE, SAR, and MAE, were part of the hardware findings that were examined. Despite some intriguing suggestions for simplifying hardware, the study is entirely concerned with design time. Because of their great effectiveness and low complexity, as shown in the research.

The FC-HBF receiver, detailed in this paper, is capable of switching between two entirely connected two-stream multi- input-multi-output (MIMO) modes at either 28 or 37 GHz, and it also has an inter-band carrier-aggregation (CA) mode that enables simultaneous single-stream operation at both 28 and 37 GHz. We create a new architecture for image-reject (IR) heterodyne beamforming that is easily reconfigurable. The front end of the Beamformer uses current-mode dual- band active combiners and coupled-resonator-based dual- band gain stages to accomplish concurrent dual-band operation [9]. This design is based on RF-domain complex- weighting and is inherently wideband. A sequence of complex-quadrature mixing steps combining Cartesian complex-weighting with image rejection makes up the down conversion phase. To further improve the suggested architecture's picture rejection, a new quadrature error detection and calibration method is also created. First implemented in RF or hybrid Beamformer, a minimum mean- square error (MMSE) beam adaptation method allows main lobe and null adaptation without requiring particular access to the Beamformer inputs of a traditional least mean-square (LMS) scheme.

Noncontiguous transmission networks and high power- efficiency requirements are obstacles to radio transmitter and power amplifier (PA) design and implementation. The nonlinear PA design makes it more likely that there will be large amounts of unwanted emissions. These emissions could make the receiver less sensitive or stop transmissions on nearby channels [10]. In order to mitigate these unwanted emissions, this research suggests a sub band digital predistortion method that is specifically tailored for low-cost devices using spectrally noncontiguous transmission methods. The suggested method aims to lower unwanted intermodulation distortions at the PA output by having a lot less processing complexity than traditional linearization methods. Also, new decorrelation-based parameter learning methods are shown and talked about. These allow adaptive monitoring of features that change over time and make parameter estimation easier to compute. The presentation of extensive simulation and RF measurement data obtained using a commercial LTE-Advanced mobile PA serves to validate the effectiveness of the proposed method. The results demonstrate that the proposed methods have the potential to provide very effective spurious component suppression.

Using wireless communication is an integral aspect of everyone's daily lives. A greater demand for wireless communication systems with more capacity, higher bit rates, and fast performance has emerged as a result of the rapid development of wireless technology. These networks can manage wireless data, video, and phone services. One efficient method to overcome this obstacle is to use a multicarrier modulation technology such as orthogonal frequency division multiplexing (OFDM). In this study, grayscale image processing is carried out using an LMS approach with a wavelet-based OFDM system. In a SISO setting, the AWGN and Rayleigh channels use the QPSK modulation methods. We compare the outcomes of this processing to those of a standard adaptive FFT-based OFDM system. In both setups, an adaptive filter is used to reduce the error and reconstruct the broadcast signal at the receiver. The computational cost of the FFT-based system is larger than that of the DWT-based system, more than [11]. If we look at the findings in terms of BER and signal-to-noise ratio (SNR), we can see that the adaptive DWT-based OFDM system outperforms the conventional adaptive FFT OFDM system.

III. REVERSIBLE WALLACE TREE MULTIPLIER DESIGN

It is becoming more important to have reversible computing architectures as the need for energy-efficient and quantum computing systems continues to increase. Due to the quantity of partial products and partial product reduction techniques, a typical Wallace tree multiplier will need more logic space and fan-out. In this proposal, we provide a unique technique for designing a reversible Wallace tree multiplier (RWTM) that is utilized to minimize circuit depth. The proposed approach designs the partial product circuit utilizing TG and FG gates, with TG producing the partial products and FG for fan-out. In addition, we employed a PG gate and a Feynman's block as reversible half-adders (HA) and full-adders (FA) in the adding network [12]. The major goal of this proposed strategy is to reduce circuit depth while increasing circuit speed, and the assessment results demonstrate that the proposed design is the most rapid in terms of latency.

A. Reversible logic Gates

The evaluation of reversible circuits involves assessing many criteria, including the number of gates, the quantity of constant inputs, the number of garbage outputs, the latency, and the hardware complexity. Outputs that are not used for further calculations are referred to as garbage outputs. A digital logic system typically includes common logic gates such as AND, OR, NAND, NOR, EXOR, and EX-NOR

gates. However, these gates do not possess the functionality of reversible logic. Reversible logic, on the other hand, employs logic gates such as PERES Gate, HNG Gate, TOFFOLI Gate, FEYNMAN Gate, FREDKIN Gate, TSG

Gate, and SGG Gate. These reversible logic gates are capable of performing multiple operations. In the context of arithmetic operations like full addition and full subtraction, reversible logic gates are utilized. Although the structures of these addition and subtraction operations may differ, their functionality remains the same. Examples of reversible gate-based full adders include PERES Gate Full Adder, HNG Gate Full Adder, TOFFOLI and FEYNMAN Gate Full Adder, FREDKIN Gate Full Adder, FREDKIN and FEYNMAN Gate Full Adder, TSG Gate Full Adder, and SGG Gate Full Adder [13]. This study will implement and overcome all the reversible logic full adders and identify the most efficient full adders in the reversible logic architecture. Fig. 1 illustrates the proposed design of a reversible logic full adder using FG and PG gates, and the truth table for the proposed full adder architecture is shown in Table 1.

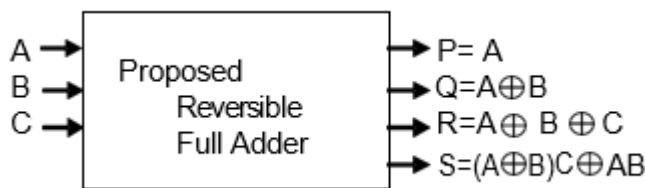


Figure 1: Block diagram of Proposed Reversible Full Adder Table 1 : Truth table for Proposed Reversible Full Adder

Table 1: Truth table for Proposed Reversible Full Adder

Input			Output			
A	B	C	P	Q	R	S
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	0	1	1	0
0	1	1	0	1	0	1
1	0	0	1	1	1	0
1	0	1	1	1	0	1
1	1	0	1	0	0	1
1	1	1	1	0	1	1

Table 2: Comparisons of Reversible logic Full Adders

	PERES Gate Full Adder	HNG Gate Full Adder	FREDKIN Gate Full Adder	FEYNMAN and PERES Gate Full Adder
Slice LUT	2	1	1	2
IOB Delay (ns)	7 6.150	7 6.150	6 6.150	7 6.110

Table 2 provides a comparison of the performance of three distinct reversible full adders; the proposed reversible FG and PG full adders will require a much smaller number of UT and have a significantly lesser delay.

B. Proposed Wallace Tree Multiplier

It will be necessary to make more use of the hardware in order to implement the partial product reduction method in the multiplier. Within the framework of the Wallace Tree multiplier, the method of partial product reduction is a crucial component. However, in the Wallace tree technique, which performs better with 4:2, 5:2, and 7:2 compressors, it will occupy a lesser number of stages, have a low logic size, and consume less power. This is in contrast to the conventional binary multiplier design, which achieves a partial product reduction by increasing the number of full adder (FA) and half adder (HA) circuits. Despite the fact that this proposed novelty-based architecture designs a Wallace Tree multiplier with reversible logic rather than a full adder, half adder, and compressor, this proposed reversible 8x8 Wallace Tree multiplication partial product generation done with FG followed by TG gate also takes exactly 64 partial products. However, because it uses TG and FG gate interconnection, it will require less critical path delay and have lower power consumption [14]. To illustrate the architecture of the development of partial products, Figure 2 will be shown.

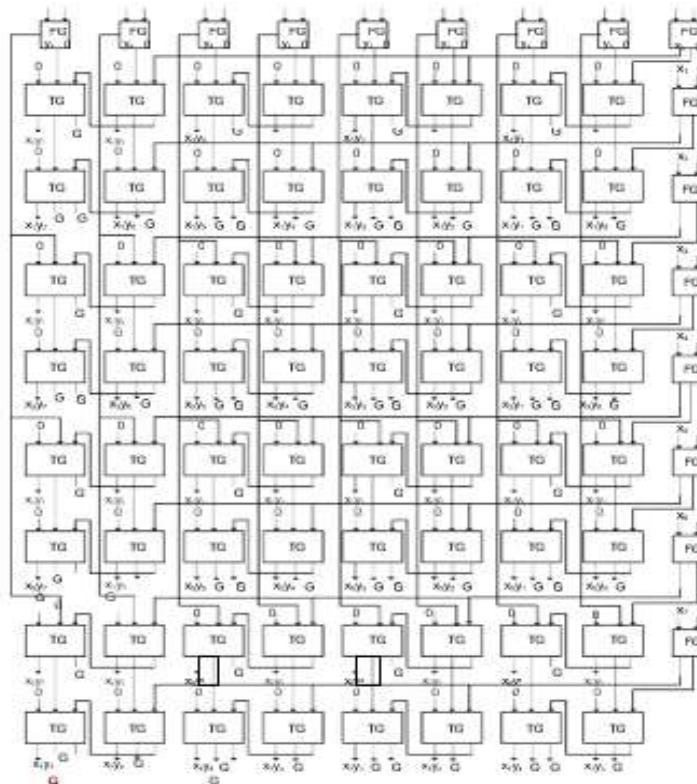


Figure 2: Partial Product generation using Reversible logic gates of FG and TG

Once we have completed the calculation of the bits that generate the partial product, we should proceed to compute their reduction of the partial product for sum outputs. In the circuit that has been presented, reversible HA and FA blocks have been built by using the PG gate and the FG gate, respectively. Figure 3 shows the simulation results, and Figure 4 is a visual depiction of the proposed Wallace tree- based multiplier.

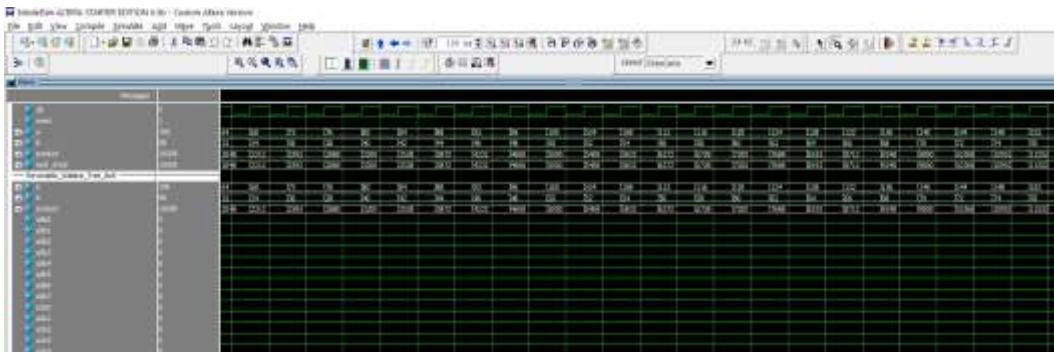


Figure 3: Simulation results of 8x8 Wallace Tree Multiplier



Figure 4: The proposed 8x8 Wallace Tree Multiplier using Reversible full adder and reversible logic gates

Table 3: Comparisons analysis of Wallace Tree Multiplier

	Conventional 1 Binary Multiplier	Wallace Tree Multiplier with 4:2 compressor and PPA	Proposed Wallace Tree Multiplier using Reversible full adders
Number of Slice LUTs	218	201	121
Number of occupied	88	73	65

Slices			
Number of IOBs	33	32	32
Delay(ns)	24.745	11.514	9.946

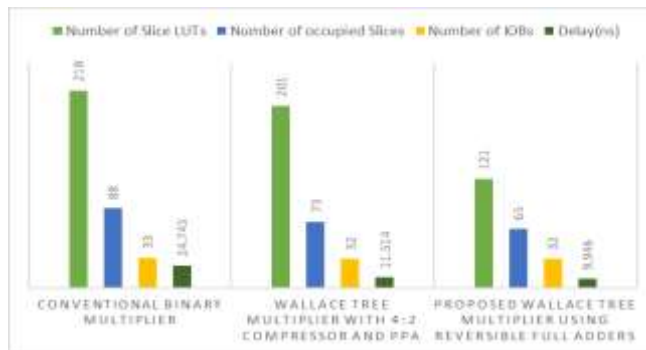


Figure 5: Xilinx results analysis graph of proposed, existing wallace tree multiplier with conventional binary multiplier

It was analyzed how the proposed multiplier stacks up against the standard binary multiplier and the standard Wallace tree multiplier design with 4:2 compressors. The comparisons and analyses of three multipliers are shown in Table 3, and the chart illustrating the Xilinx performance analysis is presented in Figure 5.

IV. PROPOSED RECONFIGURABLE DATAPATH ARCHITECTURE OF EQ-SCALABLE AF

Here we will detail the adaptive filter structure that has been suggested and the reconfigurable datapath that has been developed for adaptive filters that can scale up to EQ. All four of these modes—LMS, NLMS, PU-NLMS, and SM- NLMS—are possible with our reconfigurable adaptive filter design. Each mode makes use of a different adaptive filter technology based on LMS [15]. You can see the proposed datapath design in action in Figure 6. In this layout, you can see three control inputs denoted as Selector, γ , and Step, along with two signal inputs, which represent the reference signal $x(k)$ and the desired signal $d(k)$. Inputs γ and Step define the variables that are used for the SM-NLMS and PUNLMS filters, respectively. The Select input determines the filter mode. The SM-NLMS comparison block is also responsible for determining the optimal value of $v(k)$ for the SM-NLMS filters [16]. The input value $d(k)$ and output value $y(k)$ from the filter are used by the error block to compute the filter error.

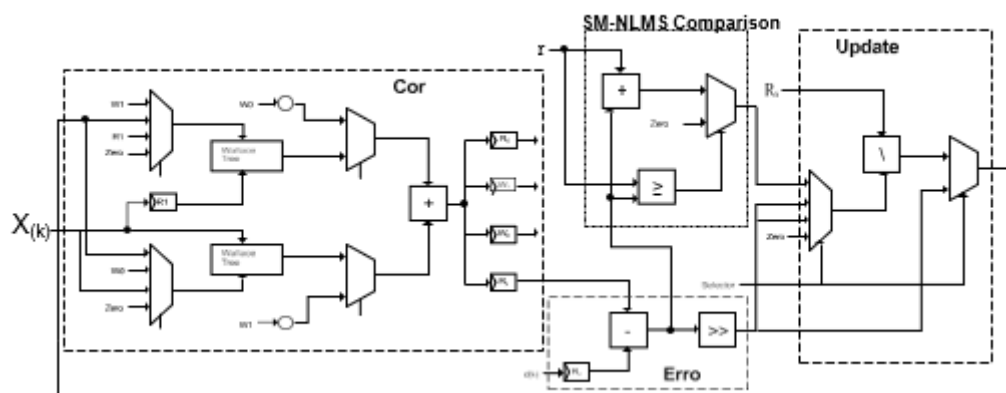


Figure 6: The reconfigurable Adaptive filter datapath proposed architecture using Wallace tree Multiplier

After the input selector determines the appropriate updating factor, the Update Control is tasked with providing it. The functions that control the SM-NLMS updating condition are a comparator block (>), two multipliers (X), two adders (+), one divider (/), and one shift operator [17]. The control of the blocks is carried out using a four-clock-cycle FSM. The procedures use a fixed-point representation due to the fact that the K least significant bits (LSBs) stand in for the fractional component. The sign value is shown as the most significant bit (MSB), and the integer component of the value is represented by the n-K MSBs that follow it.

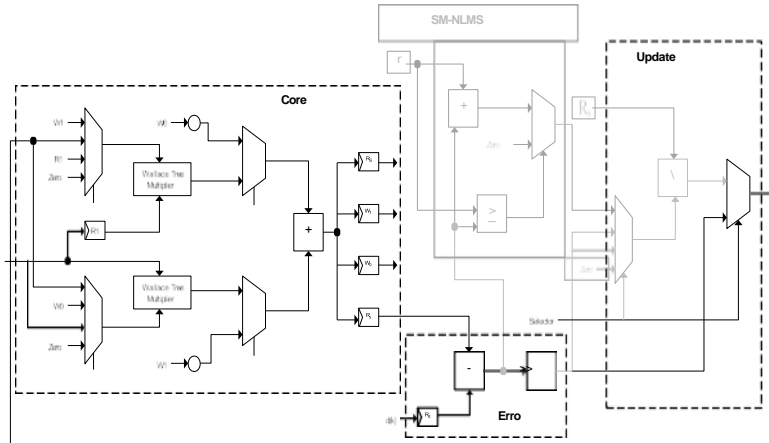


Figure 7: Operation of Reconfigurable LMS datapath architecture. The shaded blocks are un-used.

As shown in Figure 7, which illustrates the architecture's functions, the SM-NLMS contrast and revision in LMS mode, control blocks are not visible. As a recursive Wiener solution, the LMS technique decreases complexity and simplifies implementation by needing just sequential operations for multiplication and addition. With regard to the extent of the mathematical complexity they use, the FIR filter and the LMS algorithm blocks are both equivalent. Along with the FIR structure, the LMS also has an adaptive method to make sure the filter weights are updated correctly [18]. This property of the algorithm is a major factor in its extensive adoption. The formula used to construct the output LMS filter is $y(k) = W^T(k) X(k)$. Equation $w(k)$ represents the filter coefficients vector, input signal $x(k)$ is the input signal vector, and k is the discrete time order of the signal sample. By reducing the system's mean square error (MSE), which is the discrepancy between the intended signal $d(k)$ and the filter output $y(k)$, the filter weight coefficients may be determined. The filter update may be determined by using equation (1). According to [19], the adaptive algorithm step is represented by the variable μ . A ratio of 0 to 1 divided by the greatest eigenvalue of the noise signal $x(k)$'s correlation matrix R , where $\lambda(\max)$ is the highest eigenvalue, guarantees filter stability.

$$w_{(k+1)} = w_{(k)} + \mu e_{(k)} \cdot X_{(k)} \quad (1)$$

In order to get the estimate error, which is indicated as $e(k)$, one must first subtract the desired signal, which is denoted as $d(k)$, from the filter output, which is represented as $y(k)$. Therefore, $e(k)$ is equal to $d(k)$ minus $y(k)$, to put it another way. The Learning Management System (LMS) filter has a few limitations that might potentially make its implementation challenging, despite the fact that it is widely used for a wide range of solutions. There are a number of downsides, the most notable of which include a delayed resolution time, a high level of sensitivity to the noise signal intensity, and fixed coefficient updating. These are the most serious of the negatives. The low convergence speed is characterized by the fixed adaptation constant μ , which is not subject to fluctuations in the system. As a result, it offers a consistent adaption pace regardless of the changes that occur inside the system [20]. When there is insufficient conditioning of the input noise signals,

the LMS filter has a discriminating sensitivity to the power level of the input signal. This is because the LMS filter does not have adequate conditioning. As a consequence, the system's functionality can be overwhelmed by the input signal's high amplitude or large changes among samples. The adaptive algorithm updates the coefficients continuously and calculates the weight of the coefficients without applying any control. as it goes through the filter convergence process, which often results in unnecessary actions. The result of this was that further derivations from the LMS algorithm were devised in order to improve the efficiency of the technique [21].

With the exception of the design NLMS and PU-NLMS modes is shown in Fig. 8. In order to enhance the LMS filter, the NLMS changes the step-size μ according to the input power signal ($x^T(k)x(k)$). Because of this, the filter will work flawlessly even when fed signals with a lot of power. We use the formula (2) to get the weight coefficients.

$$W_{(k+1)} = W_{(k)} + \frac{\mu}{R + X^T(k)X(k)} e(k)X(k) \quad (2)$$

To guarantee stability, β must be a minimum number. Reducing the NLMS technique's hardware-related complexity is the goal of the limited update NLMS filter. This filter maintains the consistency of past results with future updates while reducing the newly calculated coefficient weight [22]. Similar to the NLMS, the weight coefficients calculation (3) takes into account a particular factor M that indicates when the coefficients need to be updated.

$$W_{(M(k+1))} = W_{(Mk)} + \frac{\mu}{\beta + X_{(Mk)}^T X_{(Mk)}} e_{(Mk)} X_{(Mk)} \quad (3)$$

Considerations such as accuracy level, system requirements, and implementation dictate the factor M value. Therefore, the correct factor M has to be found for every possible case.

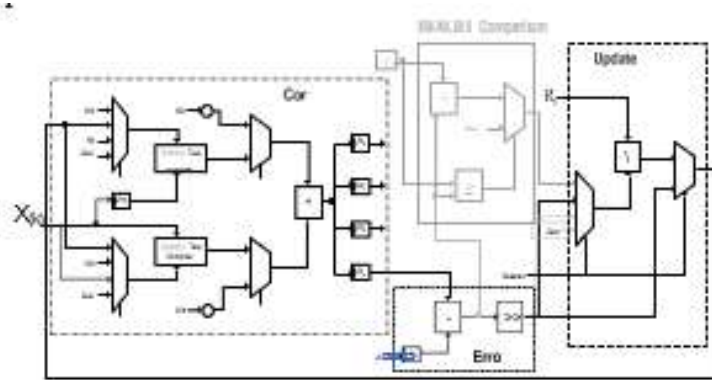


Figure 8: Operation of reconfigurable NLMS and PU- NLMS modes. The shaded blocks are un-used.

Using the whole design and the appropriate control signals, the architecture is configured for SM-NLMS mode, as shown in Fig. 9. Each mode makes advantage of the data-gating technique to stop the unused architecture—shown as shaded blocks in Figures 7–9—from switching on and off, thereby reducing their dynamic power. The goal of the set-membership normalized LMS filter is to simplify the NLMS mathematically by modifying its filter weights according to an estimate error on a specified bound, χ . If the estimated error is less than the boundary value, then any parameters and restrictions may be used [23]. When an error occurs outside of the boundary, the weights are updated by modifying the filter stages as specified in (4).

$$\mu(k) = \begin{cases} 1 & \text{if } |e(k)| > \gamma \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The NLMS filter process is identical to the error and filter output calculation procedures. Each of these processes is identical.

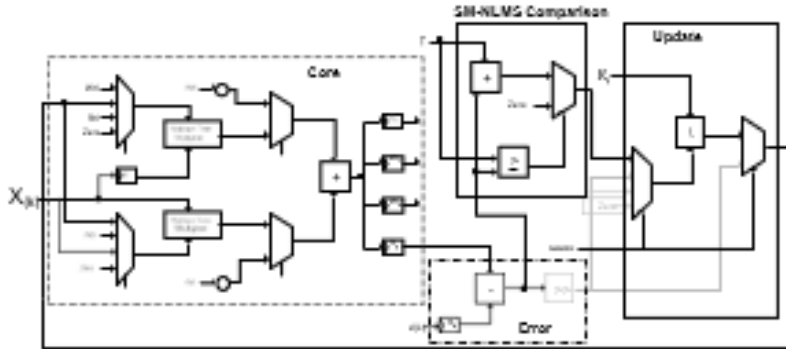


Figure 9: Operation of reconfigurable SM-NLMS mode.

The shaded blocks are un-used.

Table 4 details every property associated with each operating mode of the reconfigurable adaptive filter.

Definition of the control signal Select, as well as stages, operations, and blocks, are all part of these aspects. Note that the design prioritizes the Select indication and that you may swap modes at any moment without involving the FSM. Two multiplications and one sum are required for each mode when dealing with 2-tap to calculate the filter output $y(k)$ in all modes and α for LMS datapath architecture, respectively [24]. Results show that compared to other systems, the LMS has less mathematical complexity due to the division operation. The arithmetic operations in PU-NLMS and SM-NLMS are larger in logic size than in the NLMS, however the additional adder units are used to decrease the switching activity of the multipliers. To do this, the system's coefficients must remain constant across a significant number of cycles.

Table 4: Operating Mode of Reconfigurable Adaptive Filter

Select	Mode	Step	Process	Block
00	LMS	1	$y(k) = \sum_{i=0}^1 w_i(k) X_i(k)$	Core
		2	$e(k) = d(k) - y(k)$	Erro
		3	$\phi = \mu * e(k)$	r
		4	$w_0(k+1) = w_0(k) + \phi X_0(k)$ $w_1(k+1) = w_1(k) + \phi X_1(k)$	Core Core
01	NLMS	1	$y(k) = \sum_{i=0}^1 w_i(k) X_i(k)$ $\alpha = X^T X(k)$	Core Core
		2	$e(k) = d(k) - y(k)$ $\phi = \mu * e(k)$	Erro r
		3	$\phi = \phi / \alpha$	U.C
		4	$w_0(k+1) = w_0(k) + \phi X_0(k)$ $w_1(k+1) = w_1(k) + \phi X_1(k)$	Core Core
10	PU-NLMS	1	$y(k) = \sum_{i=0}^1 w_i(k) X_i(k)$	Core
		2	$e(k) = d(k) - y(k)$ $\phi = \mu * e(k)$	Erro

		3	$= \mu * e(k)$	r
		4	$\phi = \begin{cases} X^T X(k), & \text{if } S = M \\ 0, & \text{Otherwise} \end{cases}$	Core
			$\phi = \phi$	U.C
			α	Core
			$w0(k+1) = W0(k) + \phi X0(k)$	Core
			$w1(k+1) = W1(k) + \phi X1(k)$	
II	SM-NLMS	1	$y(k) = W^T X(k)$	Core
		2	$e(k) = d(k) - y(k)$	Error
		3	$\phi = e(k) \pm \gamma$	r
		4	$\phi = \begin{cases} X^T X(k), & \text{if } e(k) \\ 0, & \text{Otherwise} \end{cases}$	SM.C
			$\phi = \phi$	Core
			α	U.C
			$w0(k+1) = W0(k) + \phi X0(k)$	Core
			$w1(k+1) = W1(k) + \phi X1(k)$	Core

Table 5: Comparisons of Reconfigurable Adaptive filter which using Conventional Binary Multiplier and Reversible Wallace Tree Multiplier

	SMNLMS		PU-NLMS		NLMS		LMS	
	Binary Multiplier	Reversible Wallace Tree Multiplier	Binary Multiplier	Reversible Wallace Tree Multiplier	Binary Multiplier	Reversible Wallace Tree Multiplier	Binary Multiplier	Reversible Wallace Tree Multiplier
	56	88	56	89	40	39	40	39
Number of Slice LUTs	606	387	595	400	62	55	62	55
Number of Occupied Slice Registers	260	135	243	131	19	15	19	15
Number of Bonded IOBs	26	26	26	26	18	18	18	18
Delay (ns)	24.18	7.696	22.103	6.388	7.557	3.036	7.557	3.036
Power (w)	0.513	0.531	0.531	0.531	0.529	0.529	0.529	0.529

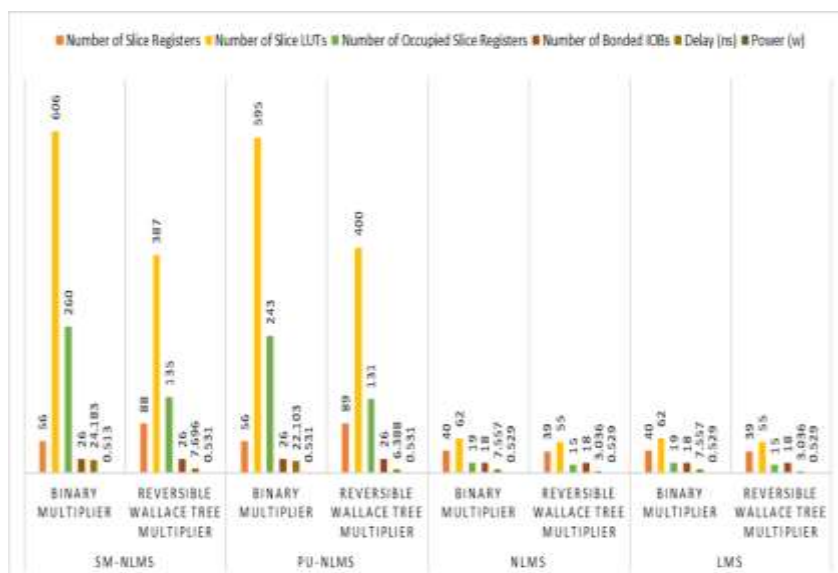


Figure 10: Comparisons analysis results of Reconfigurable Adaptive filter

V. IMPLEMENTATION REPORT OF OVERALL DESIGN ARCHITECTURE

The rest of this section details the outcomes of the idea synthesis for the reconfigurable adaptive filter architecture and how well the reconfigurable part of the implementation worked. In Table 5, a comparison was made between the four different kinds of reconfigurable adaptive filters: (a) SM- NLMS, (b) PU-NLMS, (c) NLMS, and (d) LMS. In this comparison, Verilog HDL was used for the design, Modelsim was used for simulation, and Xilinx Vertex-5 FPGA (XC5VLX50-2ff676) was used for synthesis. The creation of these filters was accomplished via the use of either a standard binary multiplier or the reversible Wallace tree multiplier proposal. Following a comparison of the two multipliers shown in Table 3, we decided to apply this multiplier to the architecture of the filter design that incorporates reconfigurable four datapaths. In this instance, the proposed multiplier by SM-NLMS will use more slice registers than the conventional technique does, but it will use less power, latency, occupied slices, LUTs, and overall efficiency [25]. In a similar manner, the PU-NLMS makes use of an 89-slice register, while the conventional approach only makes use of 56 slices. In order to complete the LUT, it will take 400, the occupied slice will take 131, and the delay will end up performing 6.388. In both the NLMS and LMS mode setups, there will be a reduction in the number of LUTs, occupied slice registers, delays, and power consumption. Presented in Figure 10 are the findings that were obtained from the examination of the reconfigurable adaptive filter.

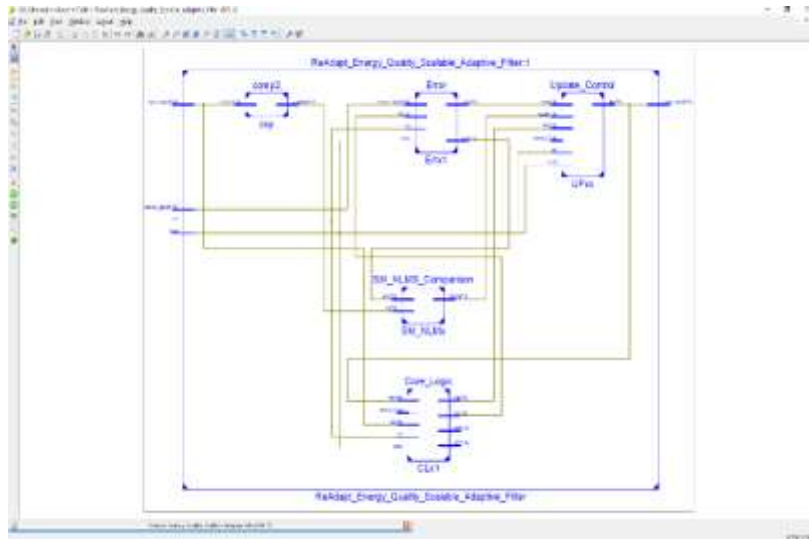


Figure 11: RTL Schematic of Reconfigurable datapath adaptive filter architecture

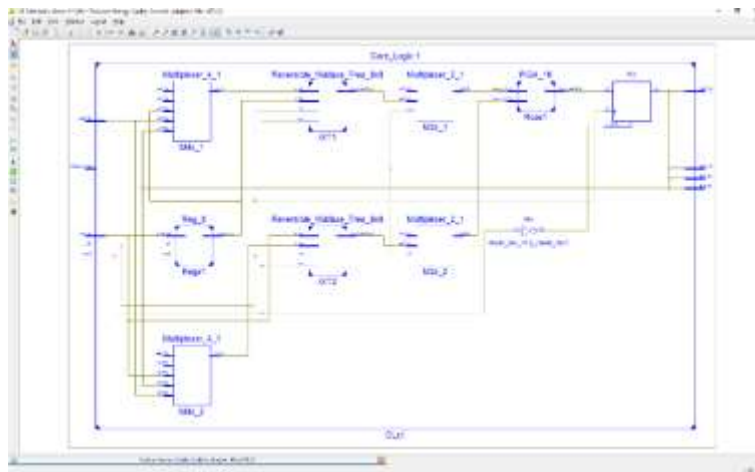


Figure 12: The Core architecture of RTL Schematic which uses reversible Wallace Tree Multiplier

Figure 11 shows the RTL Schematic of the reconfigurable datapath adaptive filter architecture, which demonstrates a novel approach. Additionally, the core architecture of the RTL Schematic, which includes a reversible wallace tree multiplier, can be found in Figure 12. Furthermore, the simulation module input sequence of this reconfigurable top module architecture, which is given with a 5G input sequence, is expected to provide the highest data rates, lower latency, and enhanced connectivity. The input sequence of 5G, which encapsulates the signal that is being conveyed, sets off on a voyage that encounters difficulties brought about by the wireless channel. These difficulties include interference, distortion, and noise.

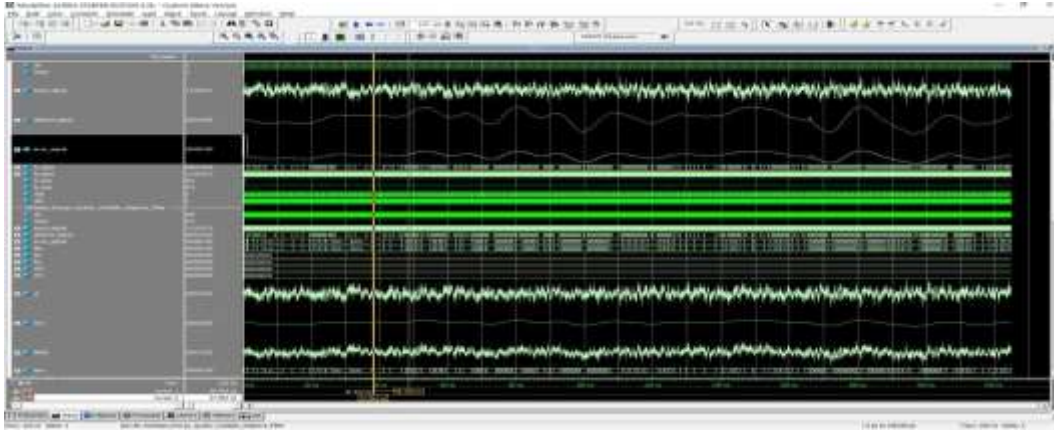


Figure 13: Simulation result analyzes of reconfigurable adaptive filter

It becomes clear that the adaptive filter is a crucial component in this current scenario because it dynamically modifies its parameters in response to the differences that exist between the signal that is received and a signal that has been established as being desired. A simulation of the proposed reconfigurable adaptive filter was performed, and the results are shown in Figure 13.

The present investigation changes LMS filters such that they use standard binary multiplications, Wallace trees, and carry look-ahead adders instead of their original multipliers and adders. Approximate distributed arithmetic circuits may be used in LMS filters after this. Using a previously published design—a parallel version of an LMS filter—the authors drew heavily on computational operators and registers to construct the circuit. The design calls for $2*W$ registers, $3*W-1$ adders, and $2*W$ multipliers when a W -tap filter is considered. It was also suggested that LMS, NLMS, and RLS filters work in tandem, with the LMS application being considered in the background. $2*W$ adders, $2*N$ memory locations, and $2*W+1$ multipliers would be the bare minimum of components for this application. The writers presented two multipliers, an adder, and a divider in relation to the NLMS example. Noteworthy is the fact that the endeavor executes every program independently. Meanwhile, we proposed an entirely sequential architecture for the FPGA-based LMS algorithm. The authors neglected to account for the much larger components of the FIR filter design and the number of clock cycles, even though the adaptive method only required a single multiplier, adder, register, and divider. Therefore, it is difficult to directly compare our work to theirs in this regard. The reconfigurable adaptive filter uses fewer components and clock cycles than a single-filter, specialized design. This is because four filters are used. If the other assessed layouts had been implemented in the Xilinx Vertex-5 FPGA with the same frequency and input vectors, our design would have achieved the greatest results in terms of processing speed, logic size, and energy savings, according to these data.

VI. CONCLUSION

A reconfigurable adaptive filter, an energy-quality-with- data-path architecture, is proposed here. This filter may dynamically pick between four different adaptive filters, such as LMS, normal architecture design of NLMS, parallel architecture design of PU-NLMS, and SM-NLMS, based on the system's requirements. In addition to reducing latency, power consumption, and system requirements, the reconfigurable adaptive filter also offers several benefits. This is on top of the fact that the proposed architecture, which employs a reversible Wallace tree design with FG and PG reversible gates, would reduce additional logic sizes. The proposed reconfigurable learning management system (LMS) design

allows for statically or dynamically reconfiguring the adaptive filtering system's complexity during runtime according to the circumstances. The beginning of battery level information or the input SNR level are two examples of the kinds of system data that could serve as references for this purpose. We found that the reconfigurable adaptive filter design is beneficial since it allows us to dynamically reconfigure the modes during runtime, which reduces energy usage while simultaneously reducing error. Numerous fields have found use for the reversible reconfigurable adaptive filter design that was suggested. These fields include biomedical signal processing, audio processing, and communication systems.

Bibliography

- [1] Binqi Yang, Zhiqiang Yu, Ji Lan, Ruoqiao Zhang, Jianyi Zhou, Wei Hong, "Digital Beamforming based Massive MIMO Transceiver for 5G Millimeter Wave Communications", *IEEE Transactions on Microwave Theory and technique*, Vol.66, No.7, July, 2018.
- [2] James Bishop, Jean Marc Chareau, Fausto Bonavitacola, European Commission Joint Research Center, E.2 Technology Innovation in Security, Italy, "Implementing 5G NR Feature in FPGA", 2018 European Conference on Networks and Communication.
- [3] Hemant Kumar, Vivek Sapru, Sandeep Kumar Jaisawal, Network R&D Team, Samsung R&D India-Bangalore, "O-RAN based proactive ANR Optimization", 2020, IEEE.
- [4] Nidhi, Albenah Mihovska, Ramjee Prasad, "Overview of 5G New Radio and Carrier Aggregation: 5G and Beyond Networks", Department of Business Development and Technology, Aarhus University, Herning, Denmark, 2020, IEEE.
- [5] M. Alioto, "Energy-quality scalable adaptive VLSI circuits and systems beyond approximate computing," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2017, pp. 127–132.
- [6] M. Alioto, V. De, and A. Marongiu, "Energy-quality scalable integrated circuits and systems: Continuing energy scaling in the twilight of Moore's law," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 8, no. 4, pp. 653–678, Dec. 2018.
- [7] F. Frustaci, S. Perri, P. Corsonello, and M. Alioto, "Energy-quality scalable adders based on nonzeroing bit truncation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 4, pp. 964–968, Apr. 2019.
- [8] B. Moons, D. Bankman, L. Yang, B. Murmann, and M. Verhelst, "BinarEye: An always-on energy-accuracy-scalable binary CNN processor with all memory on chip in 28 nm CMOS," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Apr. 2018, pp. 1–4.
- [9] A. Raha, S. Venkataramani, V. Raghunathan, and A. Raghunathan, "Energy-efficient reduce-and-rank using input-adaptive approximations," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 2, pp. 462–475, Feb. 2017.
- [10] A. Raha and V. Raghunathan, "Approximating beyond the processor: Exploring full-system energy-accuracy tradeoffs in a smart camera system," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 12, pp. 2884–2897, Dec. 2018.
- [11] G. Paim, L. M. G. Rocha, G. M. Santana, L. B. Soares, E. A. C. da Costa, and S. Bampi, "Power-, area-, and compression-efficient eight-point approximate 2-D discrete Tehebichef transform hardware design combining truncation pruning and efficient transposition buffers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 2, pp. 680–693, Feb. 2019.
- [12] R. Porto et al., "UHD 8K energy-quality scalable HEVC intra-prediction SAD unit hardware using optimized and configurable imprecise adders," *J. Real-Time Image Process.*, vol. 17, no. 5, pp. 1685–1701, Oct. 2020.
- [13] S. Çınar, "Design of an automatic hybrid system for removal of eyeblink artifacts from EEG recordings," *Biomed. Signal Process. Control*, vol. 67, May 2021, Art. no. 102543.
- [14] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y.-F. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Process. Lett.*, vol. 5, no. 5, pp. 111–114, May 1998.

- [15] A. Cheffi, M. Djendi, and A. Guessoum, "A new correlated setmembership partial-update NLMS (SM-PU-NLMSCOR) algorithm for acoustic noise reduction," in Proc. 5th Int. Conf. Electr. Eng. Boumerdes (ICEE-B), Oct. 2017, pp. 1–4.
- [16] M. Rupp, W. Kellermann, A. Zoubir, and G. Schmidt, "Advances in adaptive filtering theory and applications to acoustic and speech signal processing," *EURASIP J. Adv. Signal Process.*, vol. 2016, no. 1, pp. 1–3, Dec. 2016.
- [17] G. Akkad, A. Mansour, B. A. ElHassan, E. Inaty, R. Ayoubi, and J. A. Srar, "A pipelined reduced complexity two-stages parallel LMS structure for adaptive beamforming," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 5079–5091, Dec. 2020.
- [18] H. B. M. Ajjaiah, P. V. Hunagund, and B. R. Rao, "Adaptive filters in digital transmission based on improved LMS algorithm," in Proc. Int. Conf. Wireless Commun., Signal Process. Netw. (WiSPNET), Mar. 2016, pp. 985–988.
- [19] V. Guidotti, G. Paim, L. M. G. Rocha, E. Costa, S. Almeida, and S. Bampi, "Power-efficient approximate Newton–Raphson integer divider applied to NLMS adaptive filter for high-quality interference cancelling," *Circuits, Syst., Signal Process.*, vol. 39, no. 11, pp. 5729–5757, Nov. 2020.
- [20] A. B. La Rosa et al., "Exploring NLMS-based adaptive filter hardware architectures for eliminating power line interference in EEG signals," *Circuits, Syst., Signal Process.*, vol. 40, no. 7, pp. 3305–3337, Jul. 2021.
- [21] P. U. da Costa, G. Paim, L. M. G. Rocha, E. A. C. da Costa, S. J. M. de Almeida, and S. Bampi, "Fixed-point NLMS and IPNLMS VLSI architectures for accurate FECG and FHR processing," *IEEE Trans. Biomed. Circuits Syst.*, vol. 15, no. 5, pp. 898–911, Oct. 2021.
- [22] M. M. N. Mannan, M. A. Kamran, and M. Y. Jeong, "Identification and removal of physiological artifacts from electroencephalogram signals: A review," *IEEE Access*, vol. 6, pp. 30630–30652, 2018.
- [23] M. Z. U. Rahman, R. A. Shaik, and D. V. R. K. Reddy, "Efficient sign based normalized adaptive filtering techniques for cancelation of artifacts in ECG signals: Application to wireless biotelemetry," *Signal Process.*, vol. 91, no. 2, pp. 225–239, Feb. 2011.
- [24] A. W. Pise and P. P. Rege, "Comparative analysis of various filtering techniques for denoising EEG signals," in Proc. 6th Int. Conf. Conver. Technol. (I2CT), Apr. 2021, pp. 1–4.
- [25] K. S. Chaitanya, P. Muralidhar, and C. B. R. Rao, "Implementation of reconfigurable adaptive filtering algorithms," in Proc. Int. Conf. Signal Process. Syst., May 2009, pp. 287–291.