JJEE
Volume 11| Number 3| September 2025 | Pages 470-490

Jordan Journal of Electrical Engineering

ISSN (print): 2409-9600, ISSN (online): 2409-9619

Homepage: jjee.ttu.edu.jo

Low-Frequency and Low-Power Optimized FPGA Architecture of Adaptive Dual Threshold Filter for Real-Time Electrocardiogram Signal Denoising

Bouchra Bendahane[1*], Wissam Jenkal[2], Mostafa Laaboubi[3], Boujemaa Nassiri[4]

[1] System Engineering and Information Technology Laboratory, National School of Applied Science, Ibn Zohr University, Agadir City, Morocco.
E-mail: bouchra.bendahanne@edu.uiz.ac.ma
[2] System Engineering and Information Technology Laboratory, National School of Applied Science, Ibn Zohr University, Agadir City, Morocco.
[3] Faculty of Applied Science, Ibn Zohr University, Ait Melloul, Morocco.
[4] Interdisciplinary Applied Research Laboratory, International University of Agadir - Universiapolis, Agadir City, Morocco.

Received: Sep 29, 2025 *Revised: Oct 13, 2025* *Accepted: Oct 14, 2025* *Available online: Nov 27, 2025*

***Abstract*—** Numerous ECG denoising systems have been developed, yet many lack real-time capability and require substantial hardware and power due to the computational complexity of their underlying algorithms. The adaptive dual threshold filter (ADTF) is a recent method that has demonstrated strong denoising performance, and this paper introduces a low-power, low-hardware embedded architecture that enables real-time, high-quality ECG filtering at low processing frequencies. The proposed pipelined FPGA-based design leverages ADTF's low complexity through a non-structural implementation that significantly reduces resource usage; input data are optimized via compact unsigned binary representation to support efficient fixed-point processing; and maximum/minimum calculations are accelerated through parallel multiplexers and registers, allowing operation at only 1.44 kHz—four times the ECG acquisition rate. Denoising performance is assessed using PRD and SNR improvement, while architectural efficiency is evaluated through power, resource, and timing benchmarks, all showing substantial gains over existing ADTF architectures and other state-of-the-art systems. Overall, the results demonstrate that ADTF, combined with the proposed architecture, offers an effective, low-complexity solution for real-time ECG noise reduction and represents a suitable low-power hardware option for continuous ECG monitoring applications.

***Keywords*—** ECG; Denoising; Real-time; FPGA; Embedded; Hardware; Pipeline; VHDL; Low-frequency; Low-power.

1. INTRODUCTION

Electrocardiogram (ECG)-based cardiac disease diagnosis is a research field that has taken a wide interest from research communities. Therefore, numerous are the research papers that have been published. Advanced technologies have been integrated, such as the development of deep learning-based low-power architectures for ECG signal classification [1-3] and the development of approximate multipliers that serve digital signal processing (DSP) applications [4]. However, in a recent World Health Organization (WHO) report, coronary heart diseases are classified as the first cause of death among the ten top deadly diseases [5]. This means that more research efforts are still required from academia and industry to enhance the quality of ECG signal monitoring and cardiovascular disease diagnosis.

* Corresponding author
DOI: https://doi.org/ 10.5455/jjee.204-1759139835

ECG signal processing effectiveness is based not only on effective techniques but also on usable and noise-free ECG data. However, in reality, various kinds of noise are present when ECG data is being collected. These noises include powerline interference (PLI), baseline wander (BW), electromyogram (EMG) or muscle artifact (MA), random noise, channel noise, and electrode motion artifacts (EM) [6-8]. Consequently, and because of the sensitivity of the next processing stages to the acquired signal quality, an important part of the ECG signal literature has been focused on the denoising stage. Therefore, several techniques have been suggested, such as infinite impulse filters (IIR) and finite impulse response filters (FIR) [9-10], the empirical mode decomposition method (EMD) and its enhanced versions, namely, ensemble EMD and complete ensemble EMD [11] [12] [13] [14], wavelet-based methods [15-19], neural network-based models [20, 21], and the adaptive dual threshold filter (ADTF) [16, 17, 22]. Other works have concerned themselves with denoising quality assessment methods, such as in [23].

Given the high death percentage marked by cardiovascular diseases [5], real-time ECG signal processing is another necessity. Real-time constraint fulfillment is related to algorithm complexity. Low computational complexity algorithms present a high ability to satisfy online processing. Despite the denoising effectiveness given by the existing ECG denoising techniques, the majority of them are high-complexity methods. This means that huge amounts of hardware, time, and power are needed to complete the processing. This is not efficient, especially for real-time applications such as ECG signal analysis. In the case of EMD-based processing, this method consists of intrinsic mode functions (IMFs) extraction from the noised signal. This involves huge repetitive calculations for each IMF, where the number of these repetitions is unpredictable [21, 24]. In addition, the majority of EMD-based techniques reject the lower IMFs when reconstructing the ECG signal, so significant information is lost [17]. Unlike IIR-based filters that present non-linearity of phase response, FIR-based filters can be built to provide a linear phase across the frequency range of concern [17]. FIR-based algorithms deal with known noise that is outside the analyzed ECG frequency range. This can affect useful low-frequency components [17]. In addition, such filters are characterized by their longer delay. So special design aspects are needed, which complicate the design [7]. Conventional denoising proposed by the different wavelet-based methods involves a soft or hard thresholding operation through multiple decomposition/reconstruction levels [8]. Selecting the appropriate mother wavelet, the decomposition level, and the threshold type increases the processing complexity [17]. In the case of neural network-based works, the variety of reference signals as well as the contained noise used for the training step limit the efficacy of these approaches to this variety [17]. Comparing the filtering results given by these methods to their high complexity limits their use, especially for real-time processing [17].

Thanks to the advancement in embedded electronics, various and powerful embedded system platforms are now available with different capabilities. For digital circuits, they include multi-core central processing units (CPUs), graphical processing units (GPUs), field programmable gate arrays (FPGAs), digital signal processing (DSPs), and heterogeneous architectures such as system-on-chip (SoC) FPGAs [25]. For analog circuits, we find the field programmable analog arrays (FPAA) [26, 27]. These embedded architectures ensure pipeline-based and parallel-based computing for performant data processing. On the basis of these architectures, several ECG signal denoising algorithms have been implemented [28].

In [22, 29], Jenkal et al. proposed a real-time FPGA-based hardware implementation for the ADTF algorithm. The proposed hardware solution consisted of a structural architecture of

three pipelined blocks working under loading and processing frequencies of 360 Hz and 3.6 KHz, respectively [22, 29]. Under 360 Hz, a shifting register was used for incoming data loading. This allows real-time data loading without any need for storage either before or after data processing [22]. The real-time constraint was again fulfilled with a processing frequency of 3.6 kHz. This processing frequency is ten times the loading frequency, which is sufficient to fulfill the real-time data processing. However, a 3.6 kHz frequency is larger than the amount of processing required by the ADTF, where lower frequencies can ensure such treatment. So non-useful clock cycles are gained for subsequent treatments. A few years later, a multi-core CPU-based software architecture was proposed for the ADTF by Mejhoudi et al. [30]. The introduced software architecture consisted of multiple ADTF threads applied to multiple portions of the input ECG signal. This means that multiple storages are needed for the input data initially. The architecture was implemented in the Odroid XU4, the Raspberry B3+, and in an Intel Core i5-4200M processor-based desktop using C/C++ and the parallel programming OpenMP API (Application Programming Interface). The purpose was to accelerate the processing time to meet a real-time constraint of 2.77 ms.

In [31], a hardware architecture based on the wavelet packet transform (WPT) was introduced for ECG signal filtering. The WPT was chosen thanks to the detailed information that the WPT gives through the decomposition of the approximations and details to subsequent stages [31]. The Donoho soft thresholding approach was adopted with two-level decomposition/reconstruction. The developed architecture was compared to WT-based architectures in terms of different aspects, including resource consumption and processing frequencies. In [32], a low-power hardware accelerator for detrending physiological signals was presented [32]. The adaptive max-mean-min algorithm (AMaMeMi) was used to remove BW artifacts from several physiological signals, including ECG signals. The hardware accelerator presented the hardware part of a SW/HW architecture that was implemented in a Xilinx Zynq-7000 Soc FPGA. The mode control of the filter parameter calculation was implemented in the software part [32]. Under 100 MHz and 500 Hz frequencies, the hardware part of the AMaMeMi architecture reached a power consumption of 0.7 mW and 12 mW, respectively [32]. In [33], Tripathi et al. introduced an FIR-based hardware accelerator for EMG, PLI, and BLW removal The system involves three basic FIR filters, namely, a high pass, notch, and low pass filter for BLW, PLI , and EMG removal, respectively. Different windowing functions were used for the selection of the filter's coefficients [33]. The hardware architecture was compared to other existing FIR-based accelerators in terms of hardware and power consumption, where 35 mW of dynamic power is consumed by this later. FIR-based systems involve a stability advantage. However, they lack adaptability and suffer from increased memory [33]. Chandra et al. introduced an improved architecture for the conventional LMS adaptive filter in order to improve SNR and filter convergence speed [34]. Both fixed and floating-point data were supported. The Xilinx Vertex-6 FPGA platform was used, and the architecture was validated for PLI noise. The system's performance was compared with previous versions as well as with other related literature works. In [35], an FPGA-based hardware system for the single-node reservoir computing (SNRC) and mean filter was presented for ECG signal detrending in [35]. The architecture was developed to deal with EMG and PLI noises. A 16-bit floating-point data representation was adopted. The Xillinx Artix-7 FPGA chip was used to implement the architecture where 82 mW of power consumption was reached. Recently, Janwadkar et al.

introduced an ASIC-based low-pass FIR-based digital filter [36]. The developed filter involved a hybrid-based multiplier to overcome power consumption and on-chip area challenges.

Among the efficient techniques that have been developed for ECG signal cleaning, we find the hybrid method DWT-ADTF that is based on the ADTF and the DWT [16, 17]. The DWT is a famous time-frequency decomposition process widely used in signal processing, while the ADTF is a recent adaptive ECG denoising process [16, 17, 22]. Both DWT-ADTF and ADTF are low-complexity algorithms that have been primarily developed to be embedded in hardware architectures [16, 17, 22, 29]. We are in the process of developing an FPGA-based real-time hardware architecture for the DWT-ADTF that we will publish soon. In light of the very low computational complexity of the ADTF, this algorithm can be the subject of other future hybridization techniques that target real-time applications. For this, we provide researchers with an optimized hardware architecture of the ADTF within this research work. The optimization aspects include noise removal enhancement, hardware and power consumption reduction, and processing frequency decreasing.

2. RELATED WORKKS

This section presents the ADTF method principle, highlights its position in the literature through a comparative study, and summarizes the main embedded systems given in the literature for ECG signal denoising.

2.1. ADTF Algorithm

The adaptive dual threshold filter is an ECG signal denoising technique. It is an extended version of the dual threshold median filter that has been developed for image processing [22] . In embedded systems, especially real-time systems, algorithmic complexity is a challenging constraint because high computational complexity implies high hardware requirements, high processing time, high processing frequencies, high thermal dissipation (so more cooling needs), and therefore high-power consumption. The lower the computational complexity, the more these problems can be reduced. The ADTF has been developed not only as a soft ECG denoising method but, at first, for hardware implementation purposes [22, 29]. In addition to its effectiveness, the ADTF method presents a linear computational complexity that is linked only to the input data size [22]. In contrast, for other techniques, the computational complexity is related not only to the input data size but also to other parameters such as the decomposition level (case of DWT, EMD), the number of IMFs (EMD), and the amount of used noisy signals (EEMD) [22]. Table 1 presents the average computing time of the ADTF in comparison with other basic methods of literature. The values given for the ADTF are the results of its implementation on a computer with an i5 Intel CPU, 4 GB of RAM, and a 118 GB SSD hard disk. According to these results, the ADTF requires less time to process data compared to other works.

The denoising process of the ADTF consists of a five-sample sliding window accompanied by the calculation of two thresholds, higher and lower thresholds [22]. The computation of these thresholds is based on the sliding window's average (Avr), maximum (Mx), and minimum (Mn) values [22]. Both threshold equations are given in the Eqs. (1), (2), and (3), where z is the noise signal, β is the thresholding coefficient, which is 0.1, and L is the window size, which is 5 samples. Figure 1 resumes the ADTF principle.

Table 1. Average computation time of the ADTF in comparison with other related works (s).

Input SNR	ADTF	LPF [18]	HPF [18]	EMD [18]	FDM [18]	SWT [18]
15dB	**0.08**	3.65	3.37	4.96	5.00	3.06
10dB	**0.09**	3.36	3.24	4.71	4.81	2.99
0dB	**0.07**	2.99	3.02	4.25	4.77	2.88
Avr	**0.08**	3.33	3.21	4.64	4.86	2.97

$$Avr = \frac{1}{L}\sum_{i=n}^{n+L} z\,(i) \quad (1)$$

$$HT = Avr + (Mx - Avr) * \beta \quad (2)$$

$$LT = Avr - (Avr - Mn) * \beta \quad (3)$$

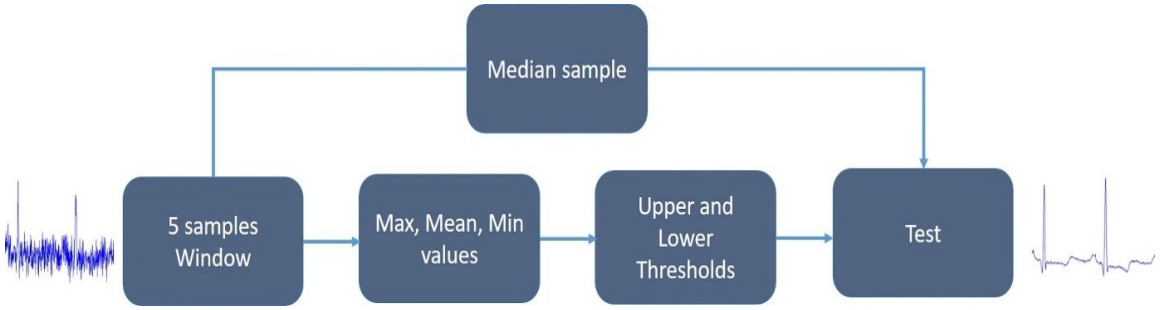

Fig. 1. ADTF algorithm principle.

Thanks to its encouraging low complexity and in the aim of improving its denoising performance, the ADTF has been the subject of hybridization with the discrete wavelet decomposition (DWT) technique. This hybridization gave birth to the DWT-ADTF algorithm [16]. The DWT-ADTF technique has significantly improved the denoising quality of the ADTF and maintained a low complexity level. In addition, an improved version of the DWT-ADTF method has been recently published [17], where promising results were achieved. This again confirms the importance of the ADTF for the real-time ECG filtering task. We are in the process of publishing an FPGA-based hardware architecture in favor of the DWT-ADTF algorithm. In this paper, we present the hardware architecture of the ADTF, which we extended and integrated into the whole DWT-ADTF hardware architecture.

2.2. Main Related Works

Because of the particular importance of the ECG signal denoising task, numerous different architectural designs have been suggested. In 2018, Jenkal et al. [22] [29] proposed a real-time hardware implementation for the ADTF algorithm using the Cyclone II DE1 FPGA device. The proposed hardware solution consisted of a structural architecture of three pipelined blocks working under loading and processing frequencies of 360 Hz and 3.6 KHz, respectively [22, 29].

A five-sample shifting register was used for incoming data receiving under 360 Hz. This allows real-time data loading without any need for data storage [22].

The real-time constraint was again fulfilled with a processing frequency of 3.6 kHz. This processing frequency is ten times the loading frequency, which is sufficient to fulfill the real-time data processing. However, a 3.6 kHz frequency is larger than the amount of processing required by the ADTF, where lower frequencies can ensure such treatment. So non-useful clock cycles are gained for subsequent treatments.

Three years later, a multi-core CPU-based software architecture was proposed for the ADTF by Mejhoudi et al. in [30]. The purpose was to accelerate the processing time to meet a real-time constraint of 2.77ms that approximately corresponds to 360 Hz, the sampling frequency of the used ECG signals (MIT-BIT Arrhythmias database). The proposed software architecture consisted of multiple ADTF threads applied to multiple portions of the ECG signal input, which means that multiple storages are needed for the input data initially. The thread number is user-chosen. The proposed architecture was implemented in the Odroid XU4, the Raspberry B3+, and in an Intel Core i5-4200M processor-based desktop using C/C++ and the parallel programming OpenMP API (Application Programming Interface). Such multi-core architecture significantly reduces the processing time. However, software architectures work under higher frequencies (GHz order) that are difficult to scale up from the system level to meet needed subprogram frequencies and consequently reduce power consumption.

In [31], wavelet-based hardware architecture was introduced for ECG signal denoising. The authors used the wavelet packet transform (WPT) instead of the standard wavelet transform. The WPT was chosen thanks to the detailed information that the WPT gives through the decomposition of the approximations and details to subsequent stages [31]. Db2 and db3 wavelet mothers were used with two-level decomposition/reconstruction and the Donoho soft thresholding process was adopted. The developed architecture was compared to WT-based architecture in terms of different aspects, including resource consumption and processing frequencies.

In 2020, a low-power hardware accelerator for detrending physiological signals was presented [32]. The hardware accelerator involved the adaptive max-mean-min algorithm (AMaMeMi) that was developed to remove BW artifacts from several physiological signals, including ECG signals. The MaMeMi filter presented the hardware part of a SW/HW architecture that was implemented in a Xilinx Zynq-7000 SoC FPGA. The software part targeted the mode control of the filter parameter calculation [32]. With an input SNR range of -18 dB to 9.37 dB using the MIT-BIH Arrhythmias and Noise Stress Test databases, the AMaMeMi architecture reached an improved SNR range of 0.78 dB to 19.38 dB [32]. Under 100 MHz and 500 Hz frequencies, the hardware part of the AMaMeMi architecture reached a power consumption of 0.7 mW and 12 mW, respectively [32]. In the same year, Tripathi et al. introduced a hardware accelerator for EMG, PLI, and BLW removal in [33]. The system proposed by the authors is a FIR-based digital filter. This later involves three basic FIR filters, namely, a high pass, low pass, and notch filter for BLW, EMG, and PLI removal, respectively. The filter coefficients selection was based on the use of different windowing functions [33]. The hardware architecture was generated using a system level integrated development environment for FPGAs [33]. The proposed system was compared to other existing FIR-based accelerators in terms of hardware and power consumption, where 35 mW of dynamic power is consumed by this later. The FIR-based systems involve a stability advantage. However, they lack adaptability and suffer from increased memory [33].

Chandra et al. proposed improved architectures for the conventional LMS adaptive filter in order to improve SNR and filter convergence speed [34]. The Xillinx Vertex-6 FPGA platform was used, and both fixed and floating-point data were supported. The architecture was validated for PLI noise in comparison with previous versions as well as with other related literature works in terms of the output SNR and resource consumption. In 2022, an FPGA-based hardware system for the single-node reservoir computing (SNRC) and mean filter was

presented for ECG signal detrending in [35]. The Xilinx Artix-7 FPGA chip was used to implement the architecture. The architecture dealt with EMG and PLI noises through a 16-bit floating-point data representation. The architecture achieved 82 mW of power consumption. Recently, an ASIC-based low-pass FIR-based digital filter was introduced by Janwadkar et al. [36]. The developed filter involved a hybrid-based multiplier to overcome power consumption and on-chip area challenges.

3. METHODS AND MATERIALS

In this section, we provide a detailed description of the proposed hardware architecture, we present the implementation material used, and we present the adopted approaches for the verification and validation of our results.

3.1. Proposed Method

The proposed ADTF architecture is a pipeline-based hardware system. It is developed through a non-structural design approach. Figure 2 presents the block schemas of this architecture. Unlike the structural modeling that was adopted for the design of the previous ADTF hardware version [22, 29], the non-structural technique helps in reducing an important number of needed resources. In hardware design, structural modeling gives more simplicity and visibility during system design and verification. However, instantiated component assembly requires more registers and signals. In contrast, this is not needed in the case of non-structural modeling since all registers and signals are directly accessible through each other. The choice between structural and non-structural modeling techniques comes back to the level of algorithm complexity. Thanks to the high simplicity of the ADTF, non-structural modeling is adopted for the design of the proposed architecture. Consequently, both power consumption and resource usage are significantly decreased.

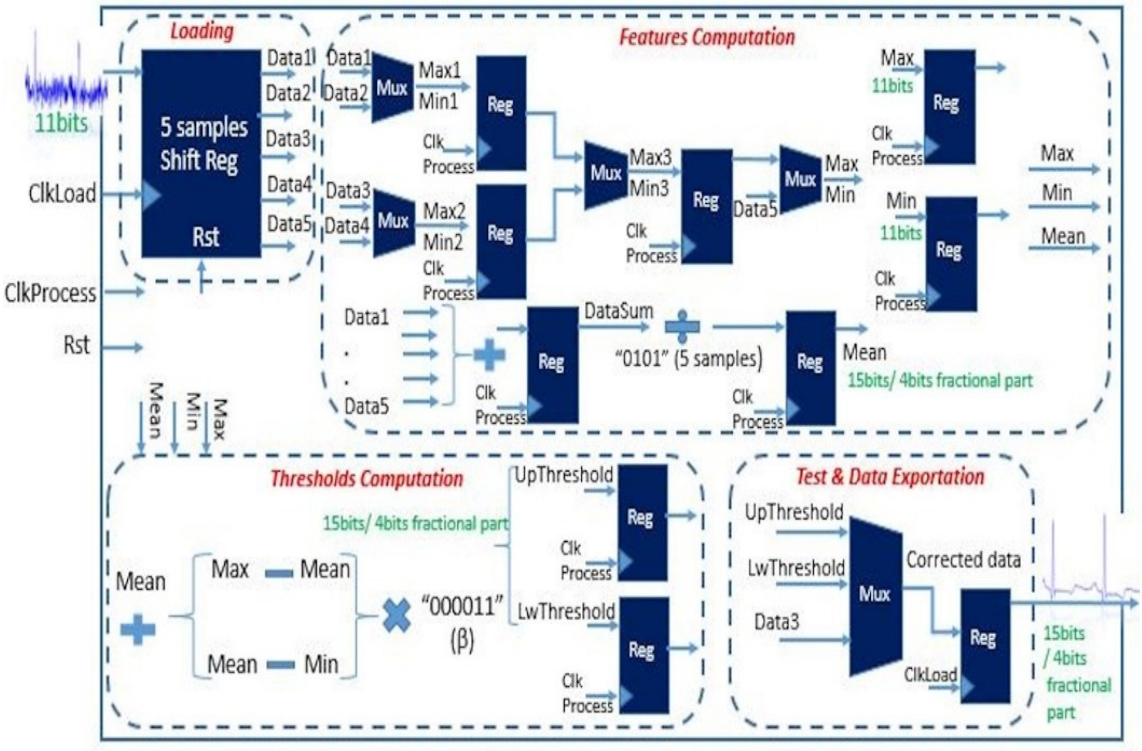

Fig. 2. Proposed ADTF based hardware architecture.

Power consumption is linked to the amount of used hardware but also to the used frequencies, where higher processing frequencies involve a higher transition rate of clock signals. Unlike the previous ADTF hardware architecture that has used only three pipeline stages, the proposed architecture is based on four pipeline blocks with three pipeline stages in the second block, which gives a total of six pipeline stages (Figs. 2, 3). Consequently, the processing frequency is reduced to 1.44 KHz, which is only four times the loading frequency (360 Hz). Therefore, the processing frequency is reduced to more than half of that used in the previous ADTF architecture (3.6 kHz).

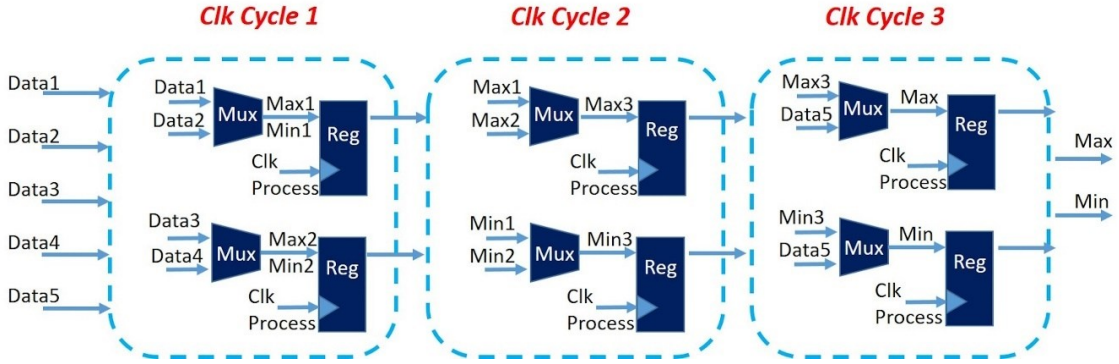

Fig. 3. Proposed hardware architecture for the maximum and minimum computing within tree-pipelined processes.

Real-world ECG data are signed real-type data. Working with such types of data in a hardware design requires huge bit handling for maintaining higher precision representation. This complicates the design process and increases the handling complexity of the signed bits. To reduce the design complexity related to data representation, ECG data is modified. Therefore, the raw ECG data is multiplied with a gain of 200, and then a baseline of 1024 is added. As a result, the data is transformed into an unsigned real with higher values, so the important data information is contained in the integer part. Integer numbers require few representative bits, and they are simple to handle in arithmetic. This modification is related to the MIT-BIH database recordings for data handling simplification [37]. This modification leads to representing data with a fixed-point representation. This later is significantly less resource- and power-consuming in comparison to a float-point data representation.

The suggested architecture is detailed within the next paragraphs of this section. The simulation outcomes for each block of the suggested architecture are displayed in Fig. 4.

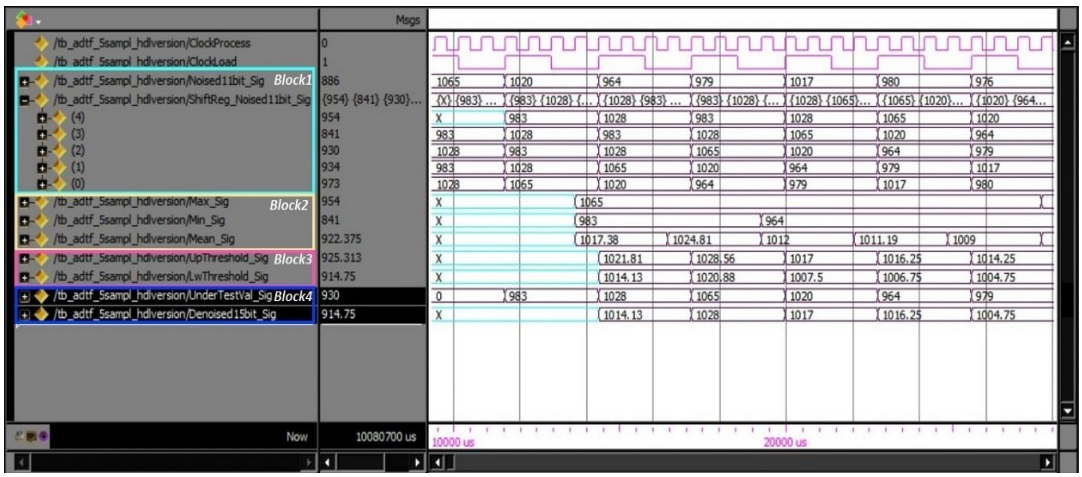

Fig. 4. Simulation results of the proposed ADTF hardware architecture.

3.1.1. Loading Module

The first block (Loading) acquires eleven bits of ECG data in the rhythm of the recording system (ClockLoad clock). The clock ClockLoad is 360 Hz since MIT-BIH Arrhythmia database signals are used in this work. The loaded data is received sample by sample through a five-sample shift register. This ensures data preparation for the next stages. Once the five samples are prepared, they are transferred to the following processing blocks in a parallel manner. Acquiring raw data through a shifting register ensures its online processing without the need for any storage before or after processing [22].

3.1.2. Features Computation Module

The Features Computation block represents the second stage of the proposed architecture. In the rhythm of the ClockProcess clock, this block calculates the minimum, mean, and maximum values of the five samples received from the loading block. The ClockProcess is 1.44 kHz, which is four times the loading clock. The maximum and minimum values are 11 bits of data with no fractional part. The mean value is 15 bits, with 12 bits for the integer part and 3 bits for the fractional part. However, the mean value will not be higher than the maximum value. Therefore, the mean value binary representation is modified so that the integer part is represented with 11 bits instead of 12 bits. The unused bit of the integer part is shifted to the fractional part. As a consequence, the mean value is 15 bits of data with 4 and 11 bits for the fractional and the integer part, respectively. This simple modification leads to more accuracy in the mean value and the related next processing operations.

The maximum and minimum values computing is a multi-cycle process and is the most clock cycles-consuming treatment of the overall ADTF. Therefore, and in order to reduce the demanded clock cycles, we have distributed this treatment onto three pipelined processes. Each process consists of two parallel multiplexers and registers (Fig. 3). The classical computing of maximum and minimum values uses a single multiplexer and register. Therefore, for N data, N-1 clock cycles are required to achieve the result. However, using the parallel multiplexers and registers, the maximum and minimum values are computed within only three clock cycles instead of four clock cycles consumed by the classical manner. This optimization was not particularly considered in the previous hardware architecture developed for the ADTF, so more clock cycles were required.

3.1.3. Thresholds Computation Module

Once the maximum, the minimum, and the mean values are obtained, the Thresholds Computation Module receives these values and calculates the two thresholds under the same processing frequency. The threshold's calculation involves a multiplication with the beta coefficient (0.1). A full binary representation of the beta coefficient needs more than forty bits. So, with the multiplication, more bits are needed, which immediately increases the hardware consumption. To overcome this issue, an approximate value is used for the beta coefficient instead of the full 0.1 value. This approximate value is chosen as having the highest approximate value and being representable with few bits. In the case of this work, the adopted approximate beta value that gives efficient computing results with the lowest hardware is "000011". The threshold values are around the mean value, so their binary representation is adjusted to meet that of the mean. Therefore, both threshold values are 15 bits of data with 11 and 4 bits for the integer and fractional parts, respectively.

3.1.4. Test and Data Exportation Module

This module represents the last processing block of the proposed architecture. It consists of the threshold-based test and the output assignment. Once the threshold values are ready, the median value of the five-sample shifting register is compared to these values. To homogenize the data representation, the fractional part of the median value is accomplished with four zero bits so that it meets the threshold binary representation. According to the comparison test, the output takes the test value (median value) itself if this value is in between both thresholds. Otherwise, the output takes the upper threshold value if the test value is higher. Otherwise, it takes the lower threshold value. To accomplish the real-time data processing, the test result is assigned to the output under the loading frequency.

3.2. Materials

The field programmable gate array (FPGA) is made of reconfigurable interconnected logical blocks. The FPGA is a high parallel, low-power consumption, and rapid prototyping device. The FPGAs are used mainly in the development of application-specific integrated circuits (ASICs) [25]. These advantages give the FPGA high flexibility in hardware architecture design and implementation, which suits a wide range of applications.

The architecture that we propose within this work is implemented on the Cyclone V Soc DE 10 FPGA device using the Intel Quartus II software tool. The Quartus II provides a complete FPGA design environment that includes tools for all FPGA design phases [38]. In this work, Quartus II 13.1 Web Edition is used, which is available for free.

The FPGA-based architecture design and implementation involve several stages, namely, the architecture description using HDL languages or HLS (High Level Synthesis) [39], behavior simulation and verification, synthesis, implementation, timing analysis and optimization, verification and validation, and finally deployment and integration. The FPGA-based architectures are designed using hardware description languages (HDLs) like VHDL and Verilog. Thanks to the rich data types as well as the concurrent and sequential process-based modeling style offered by the VHDL, this latter is adopted for the design of the proposed hardware architecture.

3.3. Proof of Concepts

The designed ADTF architecture is verified and validated through two verification methods: a functional or method in the loop (MIL) simulation and a hardware in the loop (HIL) simulation.

3.3.1. MIL Simulation

The method in the loop consists of the simulation of the system's behavior without any execution on a real hardware device. Therefore, it provides an early verification, cost efficiency, and low-complexity debugging method for embedded system development. The verification of the proposed ADTF architecture aims to assess the denoising performance of this latter in comparison with the software version of the considered algorithm as well as with some of the recent related works. As a first verification level of the proposed ADTF architecture, a functional simulation is applied, and the Mentor Graphics ModelSim [38] simulation tool is used.

The proposed ADTF architecture is simulated for a set of original ECG signals taken from the MIT-BIH Arrhythmia database [37]. The chosen signals are 100, 102, 122, 124, 203, and 233. These signals are contaminated with different noise types, namely, Gaussian and other colored noises with different input SNR values. Gaussian and colored noises are types of synthetic noises that are used to mimic the real-world conditions of data acquisition. In our case, white, blue, and pink noises are used. The white noise presents a uniformly distributed intensity across frequencies. In contrast, blue and pink noises are not uniformly distributed across frequencies, where the intensity of noise increases with frequency augmentation in the pink noise case and decreases in the case of the blue noise [22]. This enables the evaluation of the filter's robustness toward different noise distributions.

Once the corrected signals by the proposed architecture are given, they are then qualitatively and quantitatively analyzed. Each ECG signal among the six used in this work is corrupted with 5, 10, and 15 dB of input SNR for each noise case. The qualitative results are given for the 100 signal, where the noised, corrected, and original versions are plotted.

In the ECG signal denoising literature, the percentage root difference (PRD) and the signal-to-noise ratio improvement (SNR imp) are the most common metrics that researchers use to judge the filter's effectiveness. Therefore, these benchmarks are adopted for the quantification and evaluation of our proposed architecture 's effectiveness. These metrics are calculated according to the Eqs. (4) and (5), where x, z, and y are respectively the original, the noised, and the corrected signals. Lower PRD with higher SNR improvement reflect efficient denoising results. The results comparison is based on those obtained by the previous hardware architecture presented in [22] and in [29], the soft ADTF results, and other related and recent works of the literature namely [40] and [41].

$$PRD = 100\sqrt{\frac{\sum_{i=1}^{i=N}(y(i)-x(i))^2}{\sum_{i=1}^{i=N}x(i)^2}} \tag{4}$$

$$SNRimp = 10\log_{10}\frac{\sum_{i=1}^{i=N}(y(i)-z(i))^2}{\sum_{i=1}^{i=N}(y(i)-x(i))^2} \tag{5}$$

3.3.2. HIL Simulation

Unlike functional simulation, hardware-in-the-loop simulation provides realistic testing of the considered architecture since its behavior is tested on a real hardware device. Such a simulation method provides more accuracy about the system's processing and timing performances under real environmental conditions. As a second verification level of the proposed ADTF architecture, an HIL simulation is applied using the Cyclone V DE10 device, the Quartus II for synthesis and implementation, and the embedded logic analyzer Signal Tap for data acquisition. The Signal Tap tool allows real-time data acquisition from the FPGA chip according to the trigger conditions defined by the user, which helps in internal nodes and I/O pin verification and debugging [38].

4. RESULTS AND DISCUSSION

This section includes a presentation and discussion of the qualitative and quantitative results given by the MIL and the HIL simulations.

4.1. MIL Simulation

4.1.1. Qualitative Results

The proposed ADTF architecture is simulated using the Modelsim tool under loading and processing frequencies of 360 Hz and 1.44 KHz, respectively. Fig. 5 shows the Modelsim display of 100 signal denoising by the proposed architecture for 5 dB of WGN. An input SNR of 5 dB reflects intensive noise, which is clearly reduced by the proposed architecture.

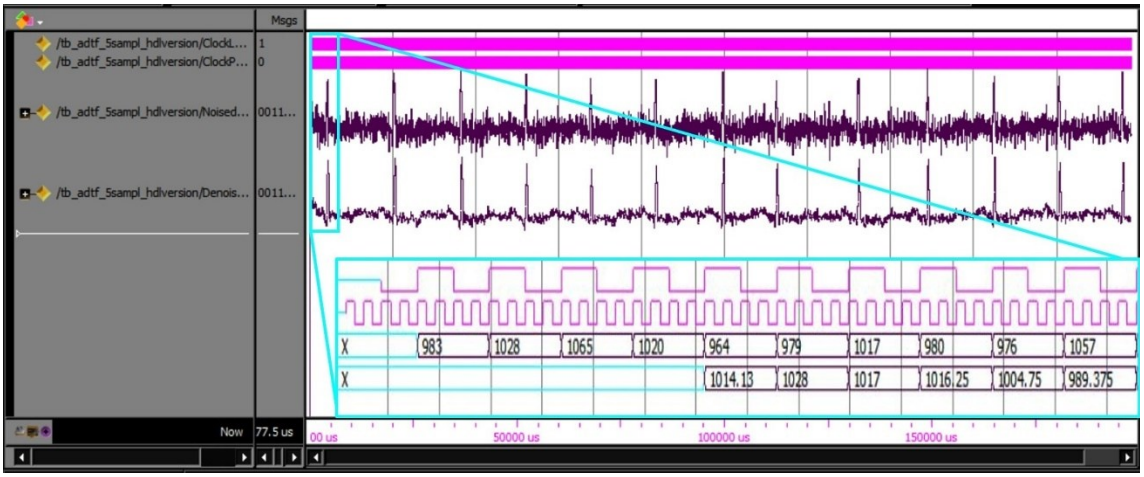

Fig. 5. Modelsim display of signal 100 denoising by the proposed ADTF architecture for 5 dB SNR of WGN.

For a clear and persuasive display, MATLAB software is used. Figure 6 presents the 100 signal denoising by the proposed ADTF architecture for 15 dB SNR of WGN in comparison with its noised and original versions. From this Figure, a high correlation is noted between the corrected and the original signals. Focusing on the QRS region, Fig. 7 shows the 100 signal denoising result in the case of WGN for different input SNR. According to these results, a high noise reduction quality is achieved by the proposed architecture. Additionally, the QRS region is well restored from the noised version of the ECG signal.

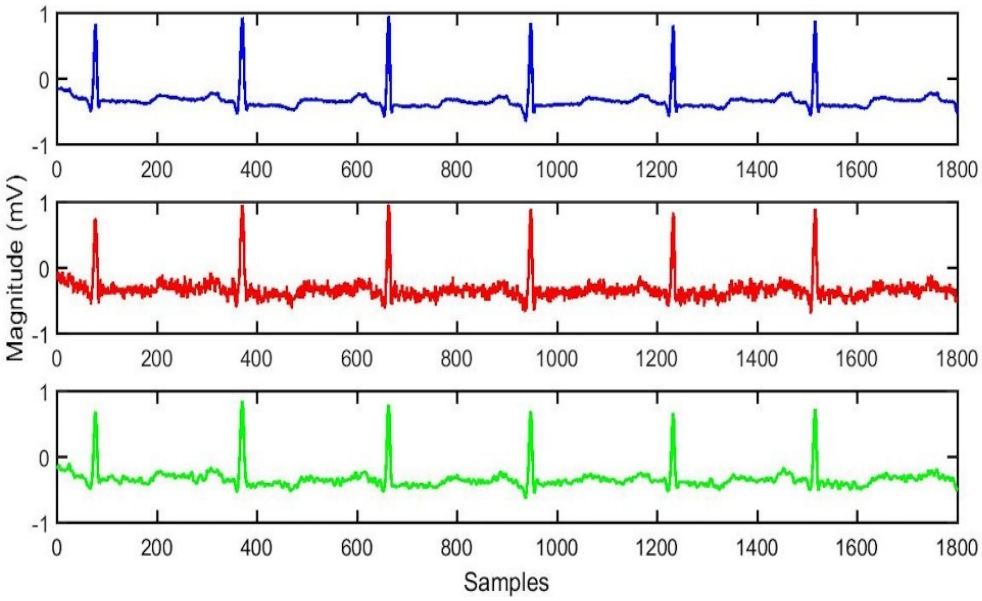

Fig. 6. MATLAB display of the 100 signal denoising by the proposed ADTF architecture for 15 dB SNR of WGN, from top to bottom: original, noised, and corrected signal.

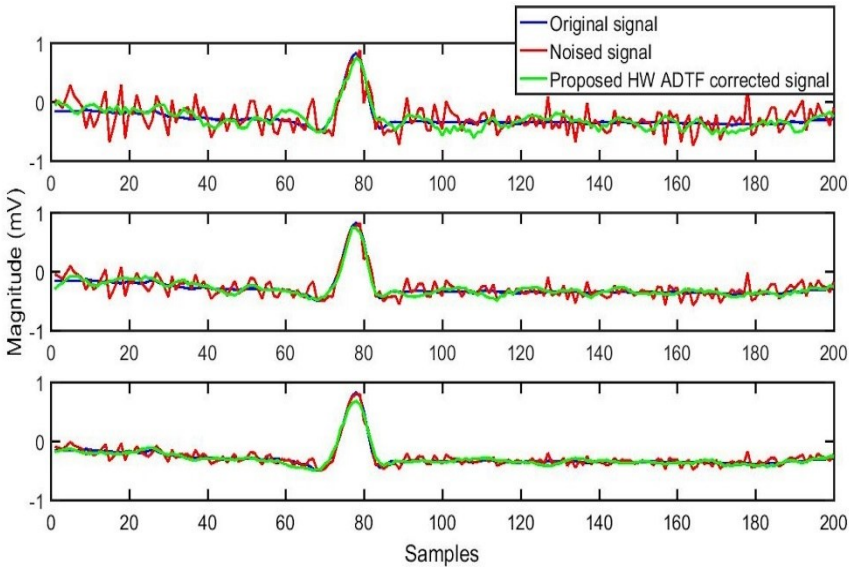

Fig. 7. MATLAB display of the 100 signal denoising by the proposed ADTF architecture in the case of WGN for different input SNRs, from top to bottom: 5 dB, 10 dB, and 15 dB case.

4.1.2. Quantitative Results

The filtering performance of the suggested architecture is quantified on the basis of the PRD, and SNR improvement benchmarks. These metrics are given for the ECG signals and in the case of the noise types previously listed. Figures 8 and 9 present, respectively, the comparison of the average SNR improvement and PRD in the case of AWGN removal.

Figure 10 presents the SNR improvement comparison in the case of 10 dB of AWGN denoising for different MIT-BIH ECG signals.

From these comparisons we can conclude that the proposed architecture gives competitive results to some of the compared works, while it outperforms others such the DWT. In addition to AWGN, the proposed architecture's effectiveness is evaluated against some colored noises. Tables 2 and 3 and give the average SNR improvement and PRD metrics comparison between the proposed HW ADTF and some related works in the case of the blue and pink noises removal, respectively. According to these statistics, the proposed architecture shows its robustness toward noise variety through its competitive outcomes.

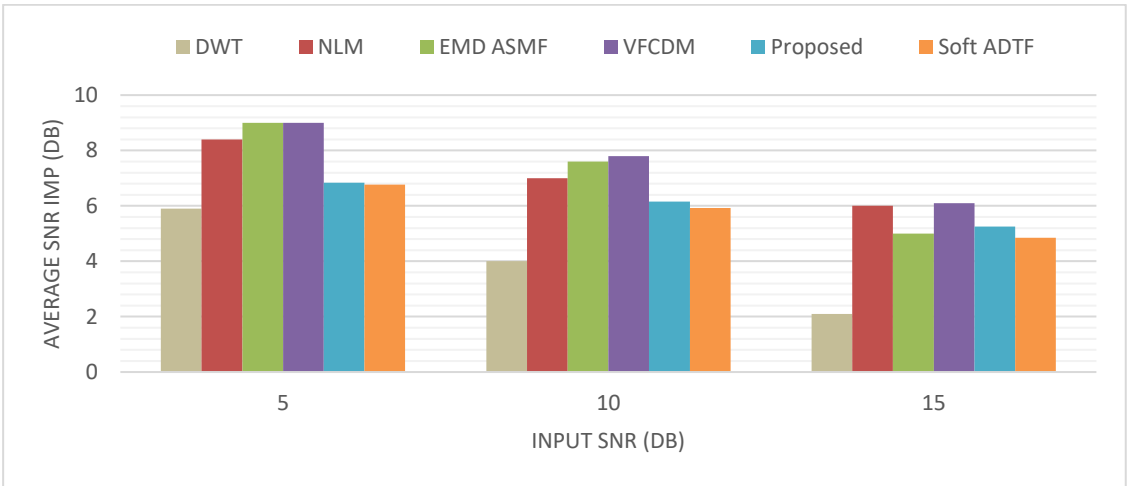

Fig. 8. Average SNR imp comparison between the proposed HW ADTF and other related works [40] in the case of WGN removal.

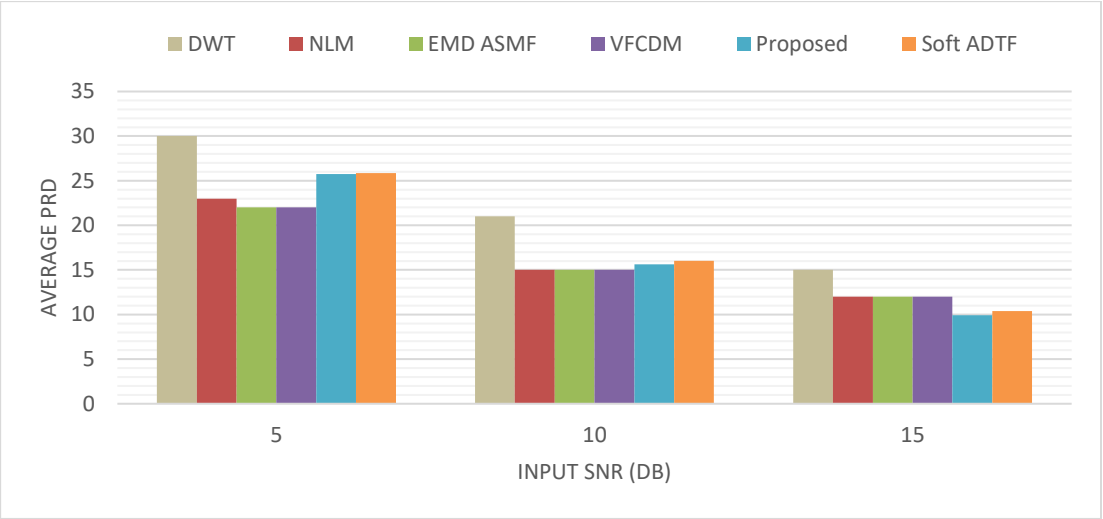

Fig. 9. Average PRD comparison between the proposed HW ADTF and other related works [40] in the case of WGN removal.

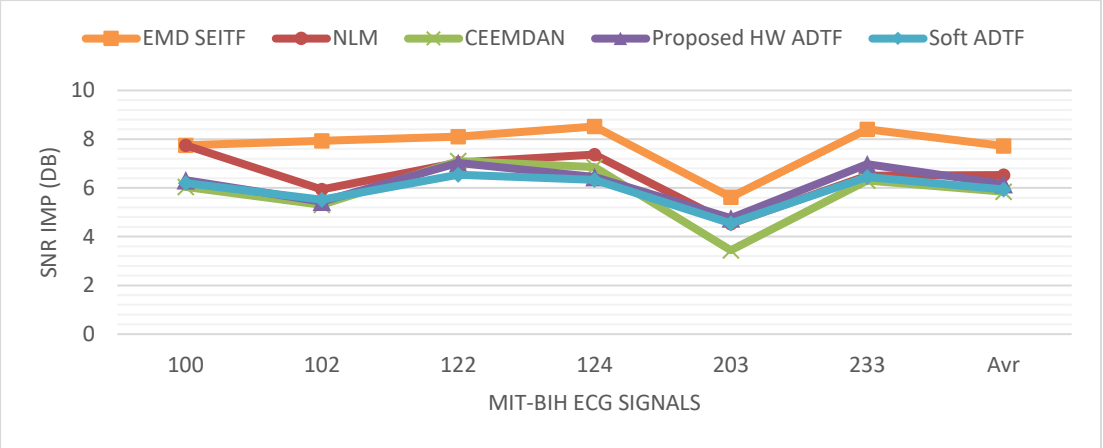

Fig. 10. SNR imp comparison between the proposed HW ADTF and other related works [41] in the case of 10 dB of WGN removal from various MIT-BIH ECG signals0

Table 2. Average SNRimp comparison between the proposed HW ADTF and other related works in the case of the blue noise removal.

	Wavelet Soft Threshold [40]	NLM [40]	EMD ASMF [40]	VFCDM [40]	Proposed HW ADTF	Soft ADTF
5 dB	7.5028	6.8032	10.934	12.7024	9.6980	9.5126
10 dB	7.5028	6.8032	10.934	12.7024	9.6980	9.5126
15 dB	5.0942	5.89	6.7723	9.3574	7.9839	7.6660

Table 3. Average PRD comparison between the proposed HW ADTF and other related works in the case of the pink noise removal.

	Wavelet Soft Threshold [40]	NLM [40]	EMD ASMF [40]	VFCDM [40]	Proposed HW ADTF	Soft ADTF
5 dB	52.2812	36.9512	47.9171	30.6806	47.5684	47.5187
10 dB	31.0743	22.6752	27.6528	21.4156	27.0378	26.9423
15 dB	18.9401	14.2122	17.242	16.3299	15.9727	16.0528

The comparison process has also considered the anterior hardware architecture given in [22] [29]. Figure 11 presents the SNR improvement comparison between the proposed and the anterior HW ADTF in the case of 5 dB of AWGN removal for different MIT-BIH ECG signals. Within this Figure, the results of the proposed architecture are superior.

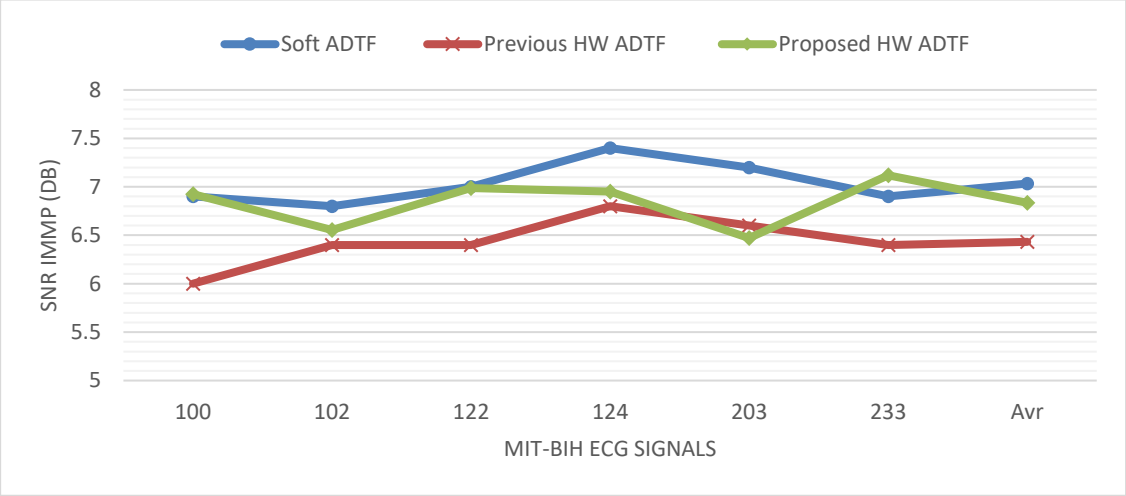

Fig. 11. SNR imp comparison between the proposed and the anterior *[22]* HW ADTF in the case of 5 dB of WNG removal from different MIT-BIH ECG signals.

4.2. HIL Simulation

4.2.1. Synthesis Results

The introduced ADTF architecture is synthesized using the Intel Quartus II. The RTL schema of the suggested hardware architecture is given in Fig. 12. The architectural performance of the suggested ADTF architecture is evaluated on the basis of the consumed power, hardware resources, and execution time. Figure 13 presents the comparison of the required logical elements between the anterior and proposed ADTF hardware architectures for different FPGA chips. Similarly, a comparison in terms of DSP and embedded multipliers is given in Table 4.

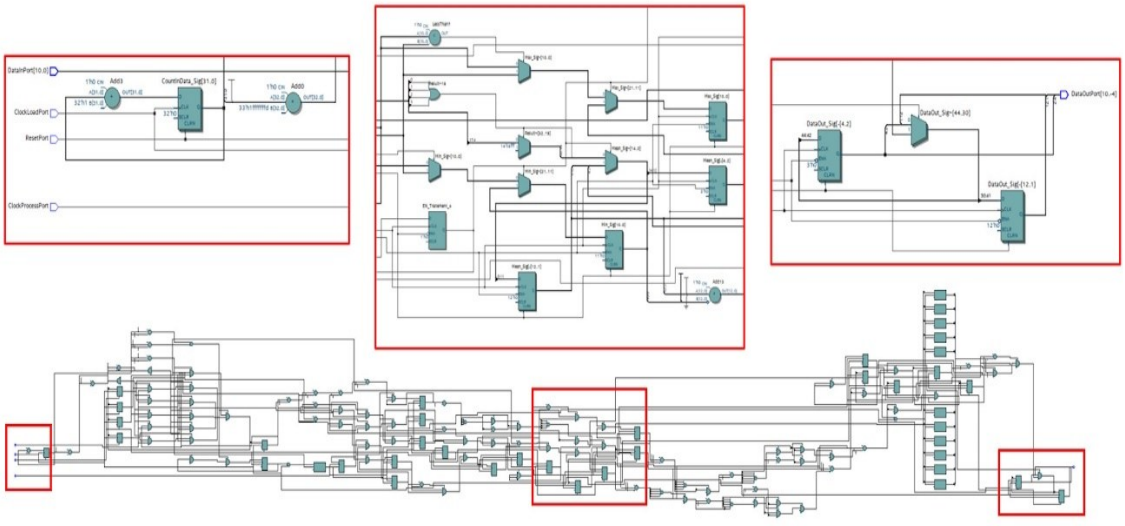

Fig. 12. RTL schema of the proposed ADTF hardware architecture given by the Quartus II.

Within these comparison results, the proposed architecture consumes fewer logical elements than is required by the anterior architecture (Fig. 13). In addition, the proposed architecture does not need any DSPs or embedded multipliers (Table 4). This is achieved thanks to our architecture's design simplicity, which leads to the use of simple multipliers and adders instead of dedicated hardware. Furthermore, under the reduced processing frequency of 1.44 kHz, the proposed architecture consumed only a dynamic power consumption of 6.70 mW to give an average SNR out of 16.26 dB. These results outperform those given by the anterior ADTF architecture and other related works, as presented in Table 5. In the same Table, we present the execution time comparison. Under the used frequencies (360 Hz and 1.44 KHz), the proposed architecture cleans the noised ECG signal within 2.8 ms of time. This means that the used processing frequency is sufficient to complete real-time processing at lower power consumption, while other architectures are unable to satisfy this compromise.

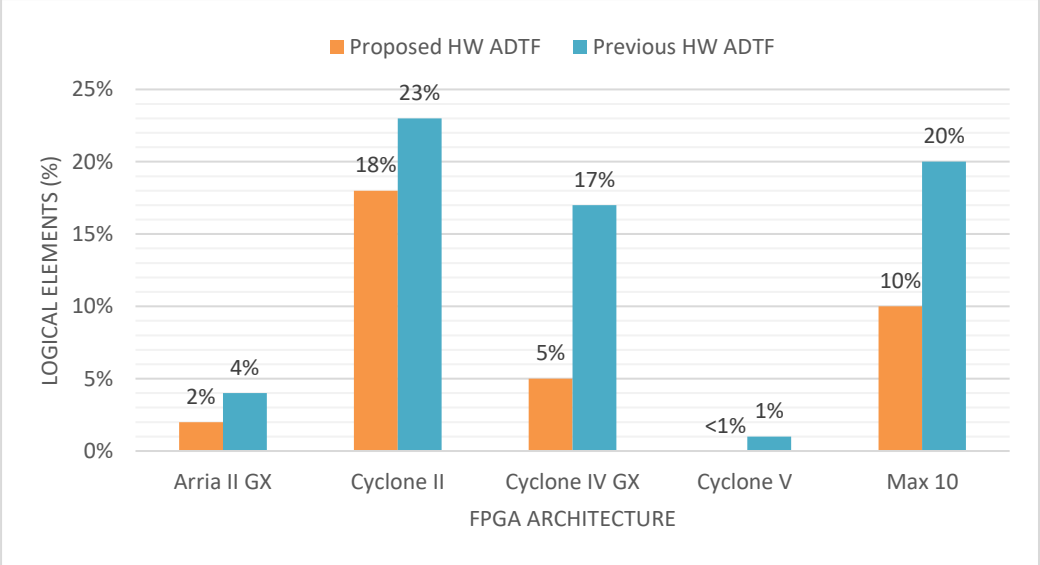

Fig. 13. Logical element consumption comparison between the previous [22] [29] and the proposed hardware ADTF architectures.

Table 4. DSPs and embedded multipliers consumption comparison between the previous and the proposed hardware ADTF [%].

Reference	Arria II GX	Cyclone II	Cyclone IV E	Cyclone V	Max 10
Proposed HW ADTF	0	0	0	0	0
Anterior HW ADTF [22, 29]	2	8	13	1	8

From all of the previous qualitative and quantitative results, we can conclude that the proposed hardware architecture gives competitive capacity against high-frequency noise corrupting ECG signals. This denoising quality is ensured in real-time, under low processing frequency, and with low hardware and power consumption. This means that additional processing can be later added, namely QRS extraction and signal classification for a complete, high-performance hardware monitoring system development. This particular advantage is not yet achieved by the majority of existing architectures due to the high computational complexity of the involved algorithms.

4.2.2. FPGA Implementation Results

The proposed ADTF architecture is implemented in a Cyclone V Soc FPGA (Fig. 14).

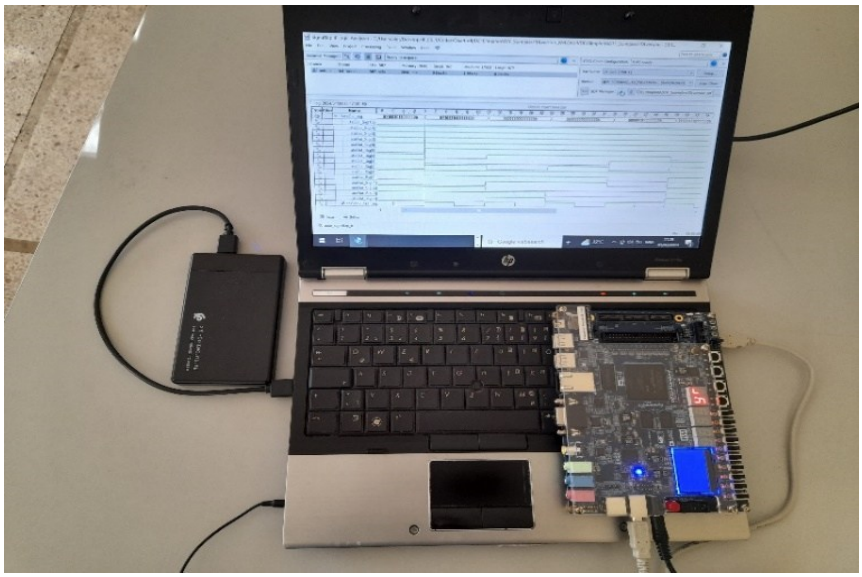

Fig. 14. Hardware implementation materials: desktop and DE10 Soc FPGA.

Table 5. Comparison of the proposed architecture with other related works in terms of power, SNRimp, SNRout, execution time and used frequencies.

Ref.	Algorithm	Architecture	Power	Frequencies	Time	SNRimp [dB]	SNRout [dB]
Propos.	**ADTF**	**hardware**	**6.70 mW**	**360 Hz and 1.44 KHz**	**2.8 ms**	**6.15**	**16.26**
[22, 29]	ADTF	hardware	-	360 Hz and 3.6 KHz	-	6.7	-
[30]	ADTF	software	5 /15.5 /90 W	1.4/2.5 GHz	2.34/7.5/0.34 ms	-	-
[31]	WT	hardware	-	177.787/195.496 MHz	-	-	-
[32]	MaxMeanMin	HW-SW	0.7/12 mW(HW part)	500Hz/ 100MHz	-	0.78	10.15
[33]	FIR filter	hardware	35 mW	-	-	-	-
[42]	CS-DAE	hardware	1.65 W	-	-	10.5	-
[43]	16 order FIR filter	hardware	32.971 mW	100.210 MHz	-	-	-
[44]	DENLMS	hardware	42 mW	-	-	-	-
[45]	FIR filter	hardware	89 mW	50 MHz	-	-	6.5813

To visualize the processed FPGA's data, the embedded logic analyzer Signal Tap is used. The Signal Tap tool is configured so that the sampling frequency to acquire the processed data from the FPGA is 360 Hz, while for the Signal Tap trigger condition, the synchronization signal of the output data is used under a basic AND trigger condition. These configurations reflect the real-time data processing condition as designed and verified in the MIL phase. Therefore, they let us verify and validate the MIL results.

Figure 15 shows the real-time acquisition through the Signal Tap graphical interface. The Figure presents the 115 signal denoising case for 5 dB of additive white noise. The Figure presents the denoised signal in comparison with its noised version, where we can see that the noise is well reduced, which validates the MIL simulation results.

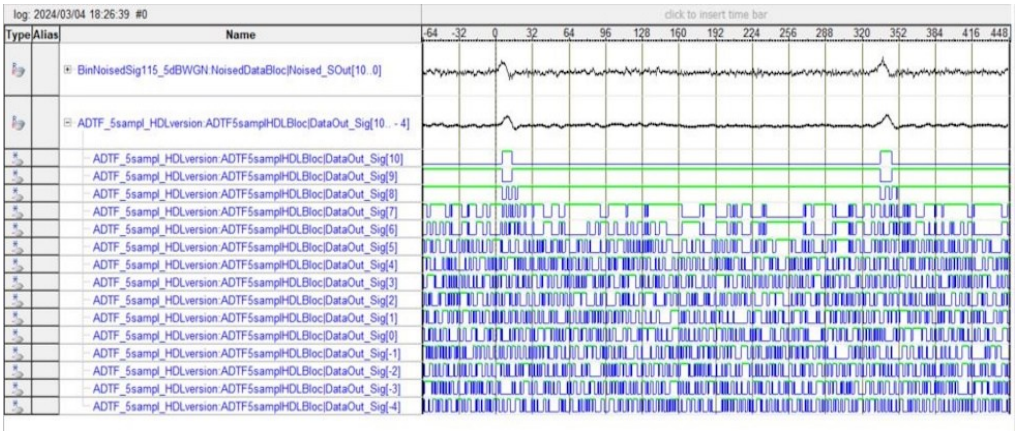

Fig. 15. FPGA implementation results of the proposed ADTF architecture in the case of the ECG signal 115 with 5 dB.

5. CONCLUSION

This work was the subject of a real-time FPGA-based hardware design and implementation of the ADTF method that is used for ECG signal denoising. Thanks to its low complexity, non-structural-based modeling was followed for the design of the proposed architecture, where significant resource needs were reduced. To reduce the required clock cycles, parallel multiplexers and registers were integrated within three pipelined processes for the maximum and minimum values computing. Consequently, a low processing frequency of only 1.44 kHz was needed. Additionally, less resources were required, and no DSPs or embedded multipliers were needed. As a consequence, an execution time of 2.8 ms and a dynamic power of only 6.70 mW were needed by our system to accomplish the ECG denoising task. Numerous ECG signals from the MIT-BIH Arrhythmias database and noise types were used to verify the architecture's performance and robustness. This was assessed through different benchmarks, including the PRD, and SNR improvement. Compared to the existing works, the proposed hardware architecture gives competitive capacity against high-frequency noise that corrupt ECG signals. This denoising quality is ensured in real-time, under low processing frequency, and with low hardware and power consumption. This means that additional processing can be later added, namely QRS extraction and signal classification for a complete, high-performance hardware monitoring system development. Therefore, the proposed architecture represents a suitable, low-power solution that can be adopted in real-time ECG monitoring systems.

The proposed ADTF architecture is extended for eleven samples processing and used in the hardware architecture that we propose for the DWT-ADTF algorithm. The ADTF architecture optimizations have had significant effects on the performance of the DWT-ADTF architecture that we will publish soon. In our future work, we aim to use float-point representation to improve the data accuracy of our proposed ADTF architecture.

Data Availability Statement: The data that supports the findings of this study are openly available in the MIT-BIH Arrythmia database of the PhysioNet website (https://physionet.org) at https://doi.org/10.1161/01.CIR.101.23.e215, reference number [37].

Acknowledgement: The corresponding author acknowledges each of the co-authors his participation in conducting the research leading to this manuscript.

Conflict of Interest: The authors declare that there is no conflict of interest regarding the publication of this article.

Funding statement: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

[1] L. Wenzhuo, C. Haoming, H. Boming, H. Yuxiang, Z. Lirong, Z. Zhuo, "Enabling on-device classification of ECG with compressed learning for health IoT," *Microelectronics Journal*, vol. 115, 2021, doi: 10.1016/j.mejo.2021.105188.

[2] C. Xuan, L. Dongsheng, L. Jiahao, W. Lai, H. Ang, L. Jianming, Z. Zhige, Z. Xuecheng, J. Quming, "Efficient hardware design of a deep U-net model for pixel-level ECG classification in healthcare device," *Microelectronics Journal*, vol. 126, 2022, doi: 10.1016/j.mejo.2022.105492.

[3] Z. Chen, L. Jian, G. Pengfei, L. Qiuping, Z. Xing, W. Xin'an, "A configurable hardware-efficient ECG classification inference engine based on CNN for mobile healthcare applications," *Microelectronics Journal*, vol. 141, 2023, doi: 10.1016/j.mejo.2023.105969.

[4] P. Pramod, K. Thottathikkulam, "Exact and approximate multiplications for signal processing applications," *Microelectronics Journal*, vol. 132, 20232, doi: 10.1016/j.mejo.2023.105688.

[5] A. Anuhya, K. Ratnam, V. Tom, P. Kumar, "An overview on state-of-the-art electrocardiogram signal processing methods: Traditional to AI-based approaches," *Expert Systems with Applications*, vol. 217, 2023, doi: 10.1016/j.eswa.2023.119561.

[6] A. Mohamed, T. Hadeel, I. Heba, N. Navaz, "ECG monitoring systems: review, architecture, processes, and key challenges," *Sensors*, vol. 20, no. 6, 2020, doi: 10.3390/s20061796.

[7] L. Shen, J. Paul, "A review of electrocardiogram filtering," *Journal of Electrocardiography*, vol. 43, no. 6, 2010, doi: 10.1016/j.jelectrocard.2010.07.007.

[8] C. Shubhojeet , T. Smita, Y. Narayan, G. Lalita, R. Kumar, "Review of noise removal techniques in ECG signals," *IET Signal Processing*, 2020, doi: 10.1049/iet-spr.2020.0104.

[9] S. Cuomo, G. Pietro, R. Farina, A. Galletti, G. Sannino, "A novel o(n) numerical scheme for ECG signal denoising," *Procedia Computer Science*, vol. 51, pp. 775-784, 2015, doi: 10.1016/j.procs.2015.05.198.

[10] L. Ming, H. Qing, X. Peng, L. Feng, H. Guang, L. Xiuling, "Constructing a guided filter by exploiting the butterworth filter for ECG signal enhancement," *Journal of Medical and Biological Engineering*, vol. 38, pp. 980–992, 2017, doi: 10.1007/s40846-017-0350-1.

[11] B. Manuel, W. Binwei, E. Kenneth, "ECG signal denoising and baseline wander correction based on the empirical mode decomposition," *Computers in Biology and Medicine*, vol. 38, no. 1, pp. 1-13, 2007, doi: 10.1016/j.compbiomed.2007.06.003.

[12] H. Weiwei, C. Nian, X. Wei, Y. Qian, Y. Zhijing, "ECG baseline wander correction based on ensemble empirical mode decomposition with complementary adaptive noise," *Journal of Medical Imaging and Health Informatics*, vol. 5, no. 8, pp. 1796-1799, 2015, doi: 10.1166/jmihi.2015.1647.

[13] A. Marcelo, S. Gastón, E. María, "Improved complete ensemble EMD: A suitable tool for biomedical signal processing," *Biomedical Signal Processing and Control*, vol. 14, pp. 19-29, 2014, doi: 10.1016/j.bspc.2014.06.009.

[14] Y. Liping, P. Zhongliang, "A new method based CEEMDAN for removal of baseline wander and powerline interference in ECG signals," *International Journal for Light and Electron Optics*, vol. 223, p. 165566, 2020, doi: 10.1016/j.ijleo.2020.165566.

[15] E. Oussama, L. Rachid, E. Khalifa, A. Abdenbi, J. Wissam, "ECG signal performance de-noising assessment based on threshold tuning of dual-tree wavelet transform," *BioMedical Engineering OnLine*, vol. 16, no. 1, 2017, doi: 10.1186/s12938-017-0315-1.

[16] J. Wissam, L. Rachid, T. Ahmed, D. Azzedine, E. Oussama, M. Fadel, "An efficient algorithm of ECG signal denoising using the adaptive dual threshold filter and the discrete wavelet transform," *Biocybernetics and Biomedical Engineering*, vol. 36, no. 3, pp. 499-508, 2016, doi: 10.1016/j.bbe.2016.04.001.

[17] J. Wissam, L. Rachid, L. Mostafa, "ECG signal denoising using an improved hybrid DWT-ADTF approach," *Cardiovascular Engineering and Technology*, vol. 15, pp. 77–94, 2023, doi: 10.1007/s13239-023-00698-8.

[18] K. Ashish, T. Harshit, K. Virender, K. Rama, K. Manjeet, "Stationary wavelet transform based ECG signal denoising method," *ISA Transactions*, vol. 114, pp. 251-262, 2021, doi: 10.1016/j.isatra.2020.12.029.

[19] S. Elouaham, A. Dliou, W. Jenkal, M. Louzazni, H. Zougagh, S. Dlimi, "Empirical wavelet transform based ECG signal filtering method," *Journal of Electrical and Computer Engineering*, 2024, doi: 10.1155/2024/9050909.

[20] B. Pingping, L. Wei, Z. Zhihua, "DeepCEDNet: An efficient deep convolutional encoder-decoder networks for ECG signal enhancement," *IEEE Access,* vol. 9, pp. 56699 - 56708, 2021, doi: 10.1109/ACCESS.2021.3072640.

[21] R. Arash, G. Aboozar, "A deep learning-based framework For ECG signal denoising based on stacked cardiac cycle tensor," *Biomedical Signal Processing and Control*, vol. 71, p. 103275, 2022, doi: 10.1016/j.bspc.2021.103275.

[22] J. Wissam, L. Rachid, T. Ahmed, E. Abdelhafid, H. Anas, E. Oussama, "Real-time hardware architecture of the adaptive dual threshold filter-based ECG signal processing," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 14, pp. 4649-4659, 2018.

[23] Z. Yatao, W. Shoushui, L. Yutao, L. Chengyu, "Performance analysis of multiscale entropy for the assessment of ECG signal quality," *Journal of Electrical and Computer Engineering*, 2015, doi: 10.1155/2015/563915.

[24] B. Bendahane, W. Jenkal, M. Laaboubi, R. Latif, "Hardware software co-design approach for ECG signal analysis," ICMISCA Conference, 2022, doi: 10.1007/978-3-031-29313-9_18.

[25] W. Jenkal, S. Mejhoudi, A. Saddik, R. Latif, "Embedded systems in biomedical engineering: case of ECG signal processing using multicores CPU and FPGA architectures," *Smart Embedded Systems and Applications*, pp. 103-145, 2022.

[26] S. MAHA, S. MAHMOUD, "Survey on field programmable analog array architectures eliminating routing network," *IEEE Access*, vol. 8, pp. 220779-220794, 2022, doi: 10.1109/ACCESS.2020.3043292.

[27] M. Diab, S. Mahmoud, "Field programmable analog arrays for implementation of generalized nth-order operational transconductance amplifier-C elliptic filters," *ETRI Journal*, vol. 42, no. 4, pp. 534-548, 2020, doi: 10.4218/etrij.2020-0104.

[28] S. Mejhoudi, R. Latif, W. Jenkal, "Advanced methods and implementation tools for cardiac signal analysis," *Advances in Science, Technology & Innovation*, 2019, doi: 10.1007/978-3-030-05276-8_11.

[29] W. Jenkal, R. Latif, A. Elouardi, S. Mejhoudi, "FPGA implementation of the real-time ADTF process using the intel-altera DE1 board for ECG signal denoising," World Conference on Complex Systems, 2019, doi: 10.1109/ICoCS.2019.8930780.

[30] S. Mejhoudi, R. Latif, A. Saddik, W. Jenkal, "Speeding up an adaptive filter-based ECG signal pre-processing on embedded architectures," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, 2021, doi: 10.14569/IJACSA.2021.0120544.

[31] P. Goel, M. Chandra, A. Anand, A. Kar, "An improved wavelet-based signal-denoising architecture with less hardware consumption," *Applied Acoustics*, vol. 156, pp. 120-127, 2019, doi: 10.1016/j.apacoust.2019.07.013.

[32] R. Mittal, A. Prince, S. Nalband, F. Robert, A. Fredo, "Low-power hardware accelerator for detrending measured biopotential data," vol. 70, p. 2000709, *IEEE Transactions on Instrumentation and Measurement*, 2020, doi: 10.1109/TIM.2020.3018235.

[33] K. Tripathi, H. Sohal, S. Jain, "Design and implementation of robust low power ECG pre-processing module," *IETE Journal of Research*, vol. 68, no. 4, pp. 2716–2722, 2020, doi: 10.1080/03772063.2020.1725660.

[34] M. Chandra, P. Goel, A. Anand, A. Kar, "Design and analysis of improved high-speed adaptive filter architectures for ECG signal denoising," *Biomedical Signal Processing and Control*, vol. 63, p. 102221, 2021, doi: 10.1016/j.bspc.2020.102221.

[35] A. Elbedwehy, A. El-Mohandes, A. Elnakib, M. Abou-Elsoud, "FPGA-based reservoir computing system for ECG denoising," *Microprocessors and Microsystems*, vol. 91, p. 104549, 2022, doi: 10.1016/j.micpro.2022.104549.

[36] S. Janwadkar, R. Dhavse, "ASIC implementation of ECG denoising FIR filter by using hybrid Vedic–Wallace tree multiplier," *International Journal of Circuit Theory and Applications*, vol. 52, no. 4, p. 1621-1646, 2023, doi: 10.1002/cta.3844.

[37] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. Ivanov, R. Mark, J. Mietus, G. Moody, C. Peng, H. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet Components of a New Research Resource for Complex Physiologic Signals," *Circulation*, vol. 101, no. 23, 2000, doi: 10.1161/01.cir.101.23.e215.

[38] *Quartus II: Introduction to the Quartus II Software (Version 10.0)*, Intel Corporation, 2010. https://www.intel.com/content/www/us/en/content-details/682200/quartus-ii-software-version-10-0-release-notes.html.

[39] B. Bendahane, W. Jenkal, M. Laaboubi, R. Latif, "HDL coder tool for ECG signal denoising," Digital Technologies and Applications Conference, 2023, doi: 10.1007/978-3-031-29857-8_75.

[40] M. Hossain, S. Bashar, J. Lazaro, N. Reljin, Y. Noh, K. Chon, "A robust ECG denoising technique using variable frequency complex demodulation," *Computer Methods and Programs in Biomedicine*, vol. 200, p. 105856, 2020, doi: 10.1016/j.cmpb.2020.105856.

[41] D. Zhang, S. Wang, F. Li, S. Tian, J. Wang, X. Ding, R. Gong, "An efficient ECG denoising method based on empirical mode decomposition, sample entropy, and improved threshold function," *Wireless Communications and Mobile Computing*, 2020, doi: 10.1155/2020/8811962.

[42] L. Chi, W. Ting, M. Salahuddin, L. He, H. Chang, C. Yu, S. Hwa, "VLSI architecture design for compact shortcut denoising autoencoder neural network of ECG signal," *IEEE Transactions on Circuits and Systems*, vol. 72, no. 4, pp. 1621-1633, 2025, doi: 10.1109/TCSI.2025.3533544.

[43] A. Mittal, A. Nandi, D. Yadav, "Comparative study of 16-order FIR filter design using different multiplication techniques," *IET Circuits, Devices & Systems*, 2017, doi: 10.1049/iet-cds.2016.0146.

[44] C. Venkatesan, P. Karthigaikumar, R. Varatharajan, "FPGA implementation of modified error normalized LMS adaptive filter for ECG noise removal," *Cluster Computing*, vol. 22, no. 5, pp. 12233–12241, 2018, doi: 10.1007/s10586-017-1602-0.

[45] P. Bhaskar, M. Uplane, "FPGA based notch filter to remove PLI noise from ECG," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 3, no. 4, 2015, doi: https://doi.org/10.17762/ijritcc.v3i4.4221.