VLSI Implementation of DENLMS Adaptive Filter for Biomedical Applications

C. Venkatesan¹, T. Thamaraimanalan² & R. Latha¹

¹ Department of Electronics and Communication Engineering, HKBK College of Engineering, Bangalore, Karnataka, India.
² Department of Electronics and Communication Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamilnadu, India.

DOI: http://doi.org/10.46431/MEJAST.2022.5109

ABSTRACT

Very Large Scale Integration (VLSI) implementation of the Delayed Error Normalized LMS (DENLMS) adaptive filter using a pipelined architecture is proposed and it can be used for biomedical applications. The proposed pipelined VLSI architecture increases the performance of adaptive filters by lowering the amount of time spent on critical path calculation. When compared to the current Error Normalized Least Mean Square (ENLMS) method for pipelining purposes, the DENLMS technique requires an additional delay element, and because of the delay element, the algorithm functions faster and more effective in low-power applications. The proposed pipelined architecture of the DENLMS filter, on the other hand, improves the efficiency while consuming less power overall. The cell leakage power reduction and total dynamic power reduction obtained by using the DENLMS filter are 31.8% and 33.5%, respectively, although the overall area of the filter has increased by 20.4%.

Keywords: DENLMS technique, Adaptive filter, Dynamic power, Pipelined architecture.

1. Introduction

Preprocessing the electrocardiogram (ECG) data is critical for many healthcare devices, but it is not an easy operation due to the signal's time-varying and non-stationary nature. Few components of ECG noise and channel noise can be described as white Gaussian noise, and their elimination is critical in biotelemetry applications. Numerous methods for removing noise from ECG signals have been studied in the literature, including filtering, averaging, wavelet transforms, and adaptive filters [1],[2].

Traditional IIR and FIR filters are inefficient at handling nonstationary and time-changing signals. As a result, several publications utilizing Wiener, Kalman, and adaptive filters concentrate on strategies for removing white Gaussian noise. To circumvent the shortcomings of standard filtering approaches, we introduce a Wiener filter for removing white Gaussian noise. Kalman filters are used in nonstationary signals to perform signal processing and filtering tasks. The Extended Kalman Filter (EKF) is one of the filtering techniques used to address the challenges associated with processing noisy ECG data [3],[4].

Because of its intrinsic ability to deal with nonstationary signals, adaptive filters are becoming increasingly useful in ECG noise suppression. Adaptive filters with the least mean square (LMS) and normalized LMS algorithms have been utilized in the past for updating coefficients because they are simple and provide good convergence performance. The output of the LMS-based adaptive filters, on the other hand, is obtained by a long critical path. Pipelined structures with delay elements are commonly used to decrease the critical route and achieve the desired sample period. A delayed error normalized LMS adaptive filter is developed in this research to achieve a high speed and low latency design with fewer computing parts [5].

The VLSI chip seen in Fig.1 is composed of several blocks and resources. Standard cells include logic gates such as the NAND, NOR, and inverters. To synthesize the design, these standard library cells are employed. The design is synthesized in this work using the Taiwan Semiconductor Manufacturing Company's (TSMC) Library [6]-[8].
The electrocardiogram (ECG) is a critical type of biological indicator because it gives critical information about the patient. The geometric structure of ECG data is frequently altered by recording and transmission procedures. A result of this distortion is caused by an insufficient diagnosis of disorders affecting the cardiovascular system. In this paper, a convertible phase size convertible trigger length delayed error generalized small noise removal technique is presented, and the phase size and tap length of the DENMS algorithm are modified simultaneously. Simultaneously, the weight update equation is altered.

Fig. 1. VLSI chip with various blocks

2. Related Works

Due to its advantages over Wiener and Kalman filters in coping with the nonstationary character of ECG signals, adaptive filters are gaining popularity in ECG noise cancellation. Adaptive filtering algorithms enable the detection of time-varying potentials and the tracking of signals' dynamic fluctuations. While adaptive filters have a significant computational cost, improvements in VLSI technology have enabled the development of low-cost high-performance chips. Low power consumption of VLSI-based biomedical products extends their life [9]-[11].

The combined area of the logic and I/O cells with routing resources defines the overall cell area in square millimeters. The operating frequency is dependent on the complexity of the design, such as the number of gates, the library files used in the design, the amount of routing resources, and the die size. Slack is predicted to determine the design's operating frequency [12]. In the construction of a timing path, slack is defined as the difference between the necessary time and the arrival time. The required period corresponds to the specified time. Arrival time indicates the amount of time required for the design to deliver the desired effects. The required time must exceed the available time. As a result, slack should always be construed positively [13]-[15].

Dynamic power, static power, and leaky power are the three basic components of power. Dynamic power is used extensively in VLSI design since it is proportional to the input switching frequency. Low power design strategies like clock gating, power gating, and multi-volt cells are proposed to reduce dynamic power. For effective design, the number of interfaces must be kept to a minimum. Any circuit implementation begins with simulation. The majority of simulation software offers schematic input, which allows for the simulation to be done once the circuit
is created using components from a cell library. The schematic input process, on the other hand, is time demanding and results in circuit complexity. To address these shortcomings, the Hardware Description Language (HDL) was created. Verilog HDL and VHDL are two well-known and commonly utilized HDLs. Both HDLs are capable of carrying out the same job, but each has distinct benefits and disadvantages. This work makes use of Verilog HDL [16],[17].

To begin, the design is programmed in Verilog HDL utilizing the Register Transfer Level (RTL) approach. If necessary, the design can also be partitioned into several smaller pieces. The next step is to verify the design's functioning by emulating the RTL code. All simulators currently available are capable of replicating both behavior-level and RTL-level development methods. After the design has been programmed at the RTL level, the design is subjected to testbench programming. The testbench's objective is to supply the design with the appropriate stimuli for functional verification [18].

Today, the majority of ASIC designs integrate Design-For-Test (DFT) circuitry to verify the chip's functionality after fabrication. Formal verification validates a design using mathematical techniques that are devoid of time and physical effects. The RTL to gate-level verification ensures that the logic has been precisely synthesized by the DC compiler. The layout tool takes care of the positioning and routing. The optimal placement position is critical for achieving improved timing and circuit complexity outcomes. Timing-driven placement is accomplished via the constraint file. The time-driven placement approach compels the layout tool to arrange the cells following the cell's timing. The layout tool inserts the clock tree into the design once the cells are placed [19].

The majority of DSP design methodologies now in use are geared at hardware synthesis and do not take into account the unique characteristics of the FPGA architecture. The authors demonstrated the ASIC design of a Decision Feedback Equalizer (DFE) based on LMS for Time Division Multiple Access (TDMA) Digital Cellular radio. To compensate for the inter-symbol interference induced by the multi-path effect, an adaptive DFE was implemented. They created an LMS architecture that takes up less space. The design, on the other hand, consumes more energy. However, optimizing for power consumption may increase device size [20],[21].

The application of adaptive filtering techniques based on LMS in VLSI may not provide the necessary performance. As a result, pipelined designs are used to boost performance by eliminating critical route computing. The DENLMS algorithm is applied utilizing pipelined latches and FPGA design. The results of HDL simulation, synthesis, and implementation are used to compare the performance of the ENLMS and DENLMS filters concerning logic cells, critical path, number of ports, and power consumption.

3. VLSI Architecture for DENLMS Adaptive Filter

The weight equation of LMS algorithms is written as

\[ y(n) = y(n-1) + \mu e(n)u(n) \]  \hspace{1cm} (1)

VLSI architectures are developed by applying pipelined interleaving and look-ahead decomposition techniques in signal processing and filtering applications. Equation (1) is transformed by applying look-ahead decomposition as.

\[ y(n) = y(n-m2) + \sum_{i=0}^{m2-1} \mu e(n-i)u(n-i) \]  \hspace{1cm} (2)
where $m_2$ is the delay elements used in the weight update loop.

In this research work, a delay element is applied in the error feedback path of the DENLMS adaptive filter. Equation (3) gives the look-ahead implementation of the delay element.

$$y(n) = y(n - m_2) + \mu \sum_{i=0}^{m_2-1} e(n - m_1 - i)u(n - m_1 - i)$$

(3)

From Equation (3), it can be observed that the overhead due to transformation is $n(m_2-1)$ adders and it can be reduced further using sum relaxation by taking $m_2^i$ terms, where $m_2^i \leq m_2$

Assume, $\mu$ is significantly small and it can be replaced with $y(n-m_2-1)$ by $y(n-m_2)$

$$e(n) = d(n) - W^T(n - m_2)u(n)$$

(4)

The DENLMS adaptive filter described in this section is evaluated following the aforementioned Equation of State (4). Fig.2 depicts a pipelined DENLMS system with 16-bit data in the pipeline. The desired output is denoted by the variable 'data1', while the FIR filter's input is denoted by the variable 'data2'. The error signal generated by the 'SUB' block is based on the data1 input and the output of the 'fir op' filter. The adaptive filtering block is responsible for generating the updated coefficients required to minimize the error of the FIR filter. The DENLMS algorithm is used to construct the adaptive filter shown in the block diagram.

**Fig.2.** Architecture of DENLMS adaptive filter for VLSI implementation

4. HDL Simulation Results and Discussion

The simulation, synthesis, and implementation findings for ENLMS and DENLMS filters are described in this section. While it is widely established that pipelining enhance filter performance, the collected data also indicate that pipelining decreases the system's power consumption. While both VHDL and Verilog HDL are capable of simulating and synthesizing digital systems, Verilog HDL is used in this study because it supports a large number of integrated circuit-related functions and libraries.
For simulation and synthesis, the Modelsim tool and the Xilinx Integrated Software Environment (ISE) are used. The implementation in the front-end circuit design is done with Virtex 5 as the target chip. The Xilinx Synthesis Tool (XST) translates the design code to a synthesizable RTL diagram with a netlist at the gate level. Once the design has been synthesized, it is ready for implementation in an FPGA device. In the Xilinx ISE environment, the synthesized design is partitioned and floor plan. The floor plan is physically placed and routed to ensure optimal execution. Xilinx ISE includes an area report, although the development of the power report was delayed. Fig.3 & Fig.4 illustrate the simulation results achieved using Modelsim and Xilinx ISE for the ENLMS and DENLMS filters, respectively. Verilog HDL programming is carried out at the gate and data flow levels. As can be seen from the figures, DENLMS filter error values are extremely low when 16-bit output values are used. DNLMS filter is not included in this section because of its extremely high error levels when compared to DENLMS and ENLMS filters.
ENLMS and DENLMS filters are implemented using VLSI in the back-end architecture. The target library is TSMC 130 nm technology, and the synthesis is performed using the Synopsys DC tool. The implementation is carried out in the prescribed manner, and the desired outcomes are attained. The overall die size for the DENLMS design with a pipeline is the same as the die size for the DENLMS architecture without a pipeline; the completed chip is illustrated in Fig.5 & Fig.6.

Fig.5. Final chip of ENLMS architecture

Fig.6. Final chip of pipelined DENLMS architecture

The proposed DNLMS design requires 40 input and output ports, 120 total nets, and 328594 logic cells. Because delay elements and pipelining latches were omitted, the ENLMS algorithm required fewer nets, logic cells, and ports. The additional memory components in the pipelined design result in a greater number of nets and logic cells in the existing technique. When compared to the ENLMS filter, the DENLMS filter increases the total area by 20.4 percent. The design's increased size is a result of a large number of logic cells.

On the other hand, slack reduction improves the filter's performance. DNLMS, ENLMS, and DENLMS all share the same ports. The ENLMS, DNLMS, and DENLMS algorithms have Slack values of 0.27, 0.31, and 0.19, respectively. The DENLMS filter eliminates 35.24 percent of the slack. The DENLMS filter has a total dynamic power of 42 mW. The ENLMS filter consumes 72 mW and the DNLMS filter consumes 67 mW, respectively. The proposed DENLMS filter has a cell leakage power of 107 W. The ENLMS filter has a cell leakage power of 154 W, while the DNLMS filter has a cell leakage power of 132 W. Table 1 summarizes the comparison findings for the ENLMS, DNLMS, and DENLMS methodologies, and Fig.7 depicts the comparison chart.
Table 1. Comparison of ENLMS, DNLMS and proposed DENLMS filters

<table>
<thead>
<tr>
<th>Considerations</th>
<th>ENLMS</th>
<th>DNLMS</th>
<th>DENLMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell Leakage Power (µW)</td>
<td>176</td>
<td>159</td>
<td>124</td>
</tr>
<tr>
<td>Maximum critical paths</td>
<td>248</td>
<td>286</td>
<td>314</td>
</tr>
<tr>
<td>Total Dynamic Power (mW)</td>
<td>84</td>
<td>71</td>
<td>64</td>
</tr>
<tr>
<td>Number of ports</td>
<td></td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>Total logic cells</td>
<td>31486</td>
<td>328594</td>
<td>335468</td>
</tr>
<tr>
<td>Number of nets</td>
<td>120</td>
<td>140</td>
<td>160</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, a variety of sophisticated LMS algorithms are used to demonstrate the VLSI implementation of adaptive filters. When it comes to performance and power consumption, the DENLMS adaptive filter surpasses the DNLMS and ENLMS algorithm-based adaptive filters by a wide margin. The ENLMS, DNLMS, and DENLMS filters have been implemented in FPGA on the Virtex 5 XC5LVX330 chip, which is a Virtex 5 FPGA. As a result of the additional delay required by pipelined DNLMS and DENLMS filters, they require more space than ENLMS filters. For its part, the DENLMS filter has a pipelined architecture that boosts performance while using less power. Even though the overall area of the DENLMS filter has increased by 20.4 percent, the total dynamic power reduction and cell leakage power reduction achieved are 31.8 percent and 33.5 percent, respectively, despite the increase in overall area. The DENLMS that has been suggested can be used to monitor and analyze ECG data.
Declarations

Source of Funding

This research did not receive any grant from funding agencies in the public, commercial, or not-for-profit sectors.

Consent for publication

Authors declare that they consented for the publication of this research work.

References


