



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** VII **Month of publication:** July 2024

DOI: <https://doi.org/10.22214/ijraset.2024.63734>

www.ijraset.com

Call: ☎ 08813907089

E-mail ID: ijraset@gmail.com

A Review of High Throughput FIR Filter Design

K P Heena

Electronics and Communication Engineering, Maharshi Dayanand University

Abstract: *Advancements in Finite Impulse Response (FIR) filter design have significantly propelled the field of Digital Signal Processing (DSP), addressing the increasing need for high-performance and efficient digital filters. FIR filters are renowned for their stability, linear phase response, and computational efficiency, making them indispensable in applications such as image filtering and frequency modulation. Their inherent advantages, including the ability to leverage Fast Fourier Transformation (FFT) techniques and minimal finite precision arithmetic errors, often make them preferable over Infinite Impulse Response (IIR) filters. Recent research efforts have focused on algorithmic and hardware modifications to optimize FIR filter architectures, employing techniques such as retiming, pipelining, and parallel processing. This paper explores various design techniques and architectures aimed at optimizing FIR filter performance. Additionally, the role of high-speed adder structures, like carry-lookahead adders, is examined for their impact on enhancing digital filter efficiency and the critical role of continuous exploration and optimization in FIR filter design to advance DSP. By addressing the demands of modern technology, these innovations foster the development of high-performance, efficient digital filters, essential for diverse applications and the ongoing progression of digital signal processing.*

Keywords: *Finite Impulse Response filter (FIR), retiming, pipelining, parallel processing, carry-look ahead adder*

I. INTRODUCTION

Digital filters play an important role in today's expanding field of Digital Signal Processing (DSP). One of the major reasons that DSP has become popular is their extraordinary performances. Though the filters are used for signal separation and signal restoration. Among them, the most used filter is Finite impulse response, FIR filters. FIR filters are widely used to filter images, modulate frequency and various other rate-applications. In signal processing, as FIR filter settles to zero in finite time, impulse response or response to any finite length input is of finite duration. Whereas in the case of infinite impulse response (IIR) filters, they may continue to respond indefinitely as they have internal feedback. Various methods are used to design optimised digital FIR filter that gives better results for different filter coefficients. It also includes advantages, like simple design implementation, minimized error function, etc[1]. As the filter length increases, the critical path length of the direct form FIR filter increases. The main advantage of transposed FIR filter is of shorter critical path as there is only one multiplication and an addition operation is involved. But, due to increased input capacitance area the performance of transposed form FIR filter is greatly affected. And also when the signal is rounded or truncated the accumulation path word length of a transpose FIR filter is larger than the input path word length. In that case, the accumulation path word length is equal to the sum of input path word length, and also the coefficient word length and the additional guard bits needed to be prevented overflow in the accumulation path. Thus, compared to direct form implementation, as its results increase register bit-width in the accumulation path giving rise to large input capacitance and large area requirement. Stability, linear phase response and improved computational efficiency are the major advantages of FIR filter over IIR filter.

In every VLSI design where there are arithmetic operations, adders are essential components and frequently form part of the critical path. A variety of architectures have been developed to improve upon the ripple carry adder (RCA) [2], primarily aimed at reducing carry propagation delay. While these advancements enhance the adder's speed, they often result in increased design area, with the carry look-ahead adder being a prime example that requires over twice the area of traditional designs. In contrast to the previous architectural revisions, the works in [3] sought improvement of RCA from the circuit point-of-view. Both works employed the dynamic circuit techniques to improve the speed. In [4], recent studies, however, have shifted focus from architectural improvements to circuit-level enhancements for the RCA. Specifically, research has employed dynamic circuit techniques to boost speed. One approach implemented a dynamic RCA (DRCA) using NP CMOS logic, while another utilized differential cascode voltage switch (DCVS) logic. Both designs demonstrated significant improvements in speed and area compared to static versions. Despite these advantages, our experiments indicate that the DRCA design [5] using NP CMOS logic, although outperforming the DCVS approach in speed, has a flaw that may lead to functional errors.

This highlights the importance of not only optimizing speed and area but also ensuring reliability in VLSI design practices. Continued exploration in this area is crucial for developing robust and efficient arithmetic operations in digital circuits. Distributed arithmetic (DA) is often favoured for its ability to eliminate the need for hardware multipliers while enabling the implementation of large filters with high throughput. Proposed by Croisier et al. in 1973, the DA algorithm facilitates digital filter implementations. However, the memory requirements for DA-based FIR filters grow exponentially with filter order N , reaching $2N$. By utilizing offset binary coding (OBC), memory size can be halved to $2N-1$ words [6]. As filter length increases, it also lengthens the critical path length of direct form FIR filters. To enhance the input data processing rate or throughput, techniques such as pipelining and parallel processing can be applied. Pipelining effectively shortens the critical path, thus improving throughput, though it does introduce additional latency[7]. High-speed adder structures also contribute to performance, with options including carry-skip adders, carry look-ahead (CLA) adders, conditional adders, and their combinations [8]. These advanced structures play a crucial role in optimizing the speed and efficiency of arithmetic operations in digital signal processing.

Digital filters are common place and an essential element of everyday electronics such as radios, cell phones, and AV receivers. Digital filters are discrete systems that perform mathematical operations on input signals to extract desired information[9]. They can be classified in various ways, but primarily, FIR (Finite Impulse Response) filters are categorized into four main types: lowpass, high pass, bandpass, and band stop filters. From a design perspective, digital filters fall into two categories: FIR and IIR (Infinite Impulse Response) filters. FIR filters offer several advantages that often make them preferable to IIR filters. Notably, they do not require feedback, which prevents the compounding of rounding errors through repeated calculations. This consistent error pattern simplifies the implementation process. Additionally, the performance and design characteristics of FIR and IIR filters differ significantly, influencing their suitability for various applications. So, while digital filters serve the same fundamental purpose of processing signals, their classification into FIR and IIR types reflects distinct operational principles and design considerations. FIR filters, with their inherent stability and straightforward implementation, are frequently chosen for applications where precision and reliability are paramount. Understanding these differences is crucial for selecting the appropriate filter for specific signal processing tasks in [10]. Under similar conditions regarding technical indicators, the output of an IIR filter includes feedback to the input, allowing it to better meet performance requirements compared to FIR filters. IIR filters typically require fewer storage units and involve fewer calculations, making them a more economical choice. However, while FIR filters exhibit strictly linear phase characteristics, IIR filters do not; the greater the selectivity of an IIR filter, the more pronounced the phase nonlinearity becomes. FIR filters employ a non-recursive structure, resulting in minimal finite precision arithmetic errors. In contrast, the recursive structure of IIR filters can lead to parasitic oscillations during operation. Additionally, FIR filters can effectively utilize Fast Fourier Transformation (FFT) techniques, which are not applicable to IIR filters. IIR filters benefit from the ability to leverage formulas, data, and tables from analog filter designs, often requiring less computational effort. Conversely, designing FIR filters typically necessitates significant computational resources, and their order may need to be high to satisfy specific design criteria. Therefore, both filter types have distinct advantages and disadvantages, making the choice between them highly dependent on the specific requirements of the application, including considerations of performance, computational efficiency, and design complexity.

Addition is the most fundamental operation in arithmetic applications, serving as the basis for other operations such as subtraction, multiplication, and division. A full adder is a straightforward circuit featuring three inputs and two outputs. Various logic styles have been employed to implement full adders at the switch level, with static CMOS logic being one of the most prominent. CMOS logic can be divided into two categories: classical design and hybrid CMOS logic. Classical design relies on a single logic style, while hybrid designs integrate two or more styles. An example of a classical approach is the Complementary CMOS (C-CMOS) full adder, which is built on a conventional CMOS structure that uses PMOS pull-up and NMOS pull-down transistors. This design offers a simple and efficient layout due to the complementary transistor pairs. However, the use of multiple large PMOS transistors leads to high input capacitance, which can adversely affect performance. Additionally, the increased size of these PMOS transistors has a direct impact on the overall area of the circuit. Despite these drawbacks, full swing performance can be achieved when this structure is utilized within more complex designs, making it a valuable option in digital circuit implementation. The Complementary Pass Transistor Logic (CPL) full adder is another example of a classical circuit, featuring a dual-rail structure comprised of 32 transistors. This design offers high speed, full swing output, and excellent driving capability, thanks to the inclusion of output static inverters and a fast differential stage formed by cross-coupled PMOS transistors. However, a significant drawback of CPL is its relatively high power consumption, primarily due to the presence of multiple internal nodes and static inverters, which contribute to static power dissipation. The key distinction between pass-transistor logic and complementary CMOS logic lies in the connection of the source side of the pass transistor network; in pass-transistor logic, it is linked to input signals rather than the supply voltage.

This design approach allows for a single pass transistor network—either PMOS or NMOS—to be sufficient for implementing the desired logic function, enhancing the circuit's efficiency. Overall, while the CPL full adder provides impressive performance in terms of speed and output characteristics, its power consumption challenges highlight the need for careful consideration when selecting logic styles for specific applications. This paper has also reviewed on Multiplier and Adder used for designing high throughput FIR filter architecture.

II. MULTIPLIER AND ADDER DESIGN

A. Multiplier

Array binary multiplier: It is a fundamental component in digital multiplication, leveraging an array of similar cells to generate incomplete products, which are subsequently summed together. In high-performance computing environments, parallel implementations are favoured to minimize computation time and enhance efficiency. Specifically, constructing an $M \times N$ array multiplier necessitates $(M-1)$ n -bit adders and $M \times N$ AND gates, emphasizing its reliance on both adders and gates for functionality. One effective strategy to enhance the speed of multiplication is the use of Booth encoding. This method reduces the number of partial products generated during the multiplication process, leading to fewer additions required compared to traditional multiplication techniques. In the context of array multipliers, the carry lookahead adder (CLA) is commonly employed. The integration of Booth encoding with the CLA not only optimizes the architecture but also significantly boosts speed and minimizes energy consumption by decreasing the number of partial products involved. In a study by B.N. Mohan Kumar[11], a novel approach called the AM-CIA-FIR filter was proposed, utilizing a Carry Increment Adder (CIA) to aggregate outputs from processing elements (PE). Experimental results indicated that this new design consumes 14.01% less FPGA resources compared to the conventional Parallel Shift Accumulator (PSA-FIR) filter. This showcases the effectiveness of using specialized adders in enhancing the efficiency of digital filters.

Vedic Multiplier: The research conducted by A. Bisoyi[12] highlights the implementation of digital multipliers based on Vedic mathematics, specifically utilizing the Urdhva Tiryakbyham algorithm. This approach has shown promising results, requiring 128 out of 232 bits for both 32-bit Vedic and traditional binary multipliers, achieving an impressive utilization rate of 55%. The Vedic method stands out due to its inherent efficiency in managing resources during multiplication.

Architectural Innovations in Vedic Multipliers: K. D. Rao's[13] work proposed two architectures for a Vedic real multiplier, grounded in the principles of the Urdhva Tiryakbyham sutra. The research also formulated an expression to calculate path delay in an $N \times N$ Vedic real multiplier, revealing that Vedic multipliers not only deliver lower path delays but also consume less power compared to their Booth counterparts, although they may use more power than array multipliers.

High-Speed Vedic Multipliers: Further exploration into Vedic multipliers was conducted by Koyel Dey[14], who designed an 8-bit digital multiplier utilizing Vedic algorithms and advanced low-power 16nm technology. Simulation results indicated that this 8-bit Vedic multiplier significantly outperforms other technologies in terms of Power Delay Product (PDP), demonstrating its superiority in speed and energy efficiency.

Expanding on Vedic Techniques: G. Challa Ram's[15] research emphasizes the design of high-speed Vedic multipliers based on 16 distinct sutras. The effectiveness of the Urdhva Tiryakbyham method was demonstrated, showing that Vedic multipliers consistently deliver lower latency compared to array multipliers. Notably, the latency reduction becomes more pronounced as the number of bits increases, underscoring the scalability of Vedic multiplication techniques. KK. Deergha Rao[16] proposed innovative architectures for 24×24 bit Vedic real multipliers intended for IEEE 754 format complex floating-point multiplication. These architectures aim to optimize mantissa multiplication within a 32-bit floating point context. Comparisons between the newly developed architecture and traditional approaches indicate a significant reduction in path delay, further cementing the efficiency of Vedic multipliers in advanced applications.

Booth Multipliers: Design and Implementation: Booth multipliers are essential components in various high-performance applications, including arithmetic logic units (ALUs), graphics processing units (GPUs), and digital signal processing (DSP) systems[17]. Their widespread use can be attributed to their ability to efficiently handle multiplication tasks, making them a popular choice in the design of integrated circuits. Booth encoding transforms the multiplier word into a radix-4 scheme, effectively halving the number of partial products generated during multiplication. This reduction leads to fewer hardware requirements, particularly in terms of adders, which is crucial for minimizing area and power consumption in integrated circuit design. The encoding process can be implemented in real time with only a modest increase in hardware complexity, allowing for a substantial reduction in the number of additions required. This balance of performance and resource efficiency has solidified the Booth multiplier's reputation in high-performance applications.

To further enhance delay and power efficiency, some researchers have employed modified Booth encoders[18]. These enhancements simplify the partial product generation circuit by removing unnecessary components, thus optimizing overall performance.

High-Speed Modified Booth Multipliers: In their work, Ravindra P. Rajput and M. N. Shanmukha Swamy[19] presented a design for a high-speed 8x8 modified Booth encoder multiplier capable of handling both signed and unsigned numbers. Their implementation utilizes carry save adders (CSAs) and carry look-ahead adders (CLA) to enhance the multiplication process. They provided various simulation results showcasing the effectiveness of the 8x8 modified Booth multiplier, demonstrating its operational efficiency in binary form.

Hybrid Architecture for Enhanced Performance: Uvaraj Subramanian and Srinivasan Alavandar[20] explored an 8x8 multiplier built on a hybrid architecture. Their research involved synthesizing and implementing the multiplier on FPGA using Verilog HDL. They compared the outputs of the conventional design with their hybrid architecture, highlighting the performance benefits of their approach.

Wallace Tree Multipliers: Himanshu Bansal, K.G. Sharma and Tripti Sharma[21] conducted a comparative study on various types of Wallace tree multipliers, focusing on reducing the complexity of conventional designs. Their analysis revealed that by decreasing the number of half adders used in the Wallace tree structure by 80%, while increasing the number of full adders, the overall complexity of the multiplier could be significantly reduced. Damarla Paradhasaradhi[22] and co-authors designed modified Wallace tree multipliers for both 8-bit and 16-bit architectures. Their study involved comparing the performance of these modified designs against conventional Wallace tree multipliers, employing both Ripple Carry Adders (RCA) and Square Root Carry Select Adders (SQRT CSLA) in their implementation.

While the original radix-2 Booth algorithm has proven effective, it is not without its drawbacks. Two primary issues include the variable number of addition or subtraction and shift operations, making parallel multiplier design cumbersome, and the inefficiency that arises when dealing with isolated 1's in binary representations. These limitations are effectively addressed by the Radix-4 Modified Booth Algorithm, which simplifies the multiplication process by reducing the number of partial products generated[23]. The Modified Booth Multiplier was first proposed by D.L. Macsorley[24] in 1961 and has since evolved into one of the most robust multiplication algorithms available for both signed and unsigned numbers. By minimizing the number of partial products, the Modified Booth Algorithm has become a cornerstone in modern multiplier design. Sukhmeet Kaur, Suman and Manpreet Singh Manna[25] implemented the Radix-4 Modified Booth Multiplier and compared its performance against the Radix-2 Booth Multiplier. Using VHDL for design and Xilinx ISE 9.1i software for simulation, they demonstrated that the Radix-4 variant offers superior speed, reinforcing its suitability for high-performance applications.

Vedic multipliers are gaining traction for their high throughput and energy efficiency, making them suitable for real-time DSP applications. As filter lengths increase, the critical path length of direct form FIR filters also grows, potentially hindering performance. To address this, researchers have leveraged pipelining and parallel processing to enhance input data processing rates. Pipelining effectively reduces critical path lengths, thereby improving throughput, although it does introduce increased latency. A combined structure that integrates both direct and transpose forms to reduce latency in FIR filters. This innovative approach underscores the ongoing efforts to enhance the efficiency and performance of digital signal processing components. Booth multipliers remain a vital tool in high-performance digital multiplication, effectively balancing speed, area, and power efficiency. The continuous advancements in their design, including modifications like the Radix-4 Booth algorithm and hybrid architectures, reflect the dynamic nature of this field. The exploration of alternative methods, such as Wallace tree multipliers and Vedic multipliers, further showcases the diversity of approaches available to engineers and researchers. As technology continues to evolve, the development of efficient multiplication algorithms and architectures will be critical to meeting the growing demands of modern computing applications.

B. Adder

In the realm of Very-Large-Scale Integration (VLSI) design, adders are fundamental components essential for executing arithmetic operations. They play a critical role in various applications, particularly in digital signal processing (DSP) systems and arithmetic logic units (ALUs). Due to their importance, optimizing adder structures for delay, area, and energy efficiency is a primary concern in modern circuit design. Adder structures that are efficient in terms of delay include carry-select adders, carry-skip adders, carry look-ahead (CLA) adders, conditional adders, and various combinations of these types. Ramkumar and Kittur replaced the dual rail RCA in CSLA with a single RCA and a binary to excess one converter, reducing area and power consumption. However, this modification did not significantly improve delay performance.

Efforts to streamline adder designs have led to the adoption of single RCA implementations to replace dual rail configurations. For example, Ramkumar and Kittur proposed utilizing a binary to excess-one converter in place of one of the dual RCAs, effectively reducing area and power without compromising delay performance. This innovative approach has paved the way for more efficient adder designs in contemporary VLSI applications. That is, presented a design of a carry-select adder that offers a simple approach to reduce the area and power of a square root carry-select adder (SQRT CSLA) architecture. Their comparative results of modified SQRT CSLA with regular SQRT CSLA are based on the power-delay product and area-delay product. Bahadori et al. introduced a modified carry-skip adder, improving the delay performance of conventional carry-skip adders by incorporating incrementation and concatenation schemes into the fundamental adder structure, although this increased design complexity[26]–[32].

Ripple Carry Adder (RCA): The ripple carry adder is one of the simplest adder designs, renowned for its ease of implementation and relatively low area requirements. However, the main drawback of the RCA lies in its linear increase in delay with respect to the number of bits, making it unsuitable for high-speed applications. The carry must propagate through each stage, resulting in significant propagation delay (T_{pd}), which is a critical factor in high-performance computing scenarios.

Carry-Select Adder (CSLA): To mitigate the propagation delay seen in RCA, carry-select adders (CSLA) were introduced. The fundamental principle behind CSLA involves generating multiple carries independently and then selecting the appropriate carry to produce the sum. This method significantly reduces the propagation delay compared to RCA. However, conventional CSLA architectures have an area overhead due to the presence of dual RCAs—one for input carry equal to zero and another for input carry equal to one. This dual-structure can be unattractive for designs that prioritize area efficiency.

Modified Carry-Select Adder: Basant Kumar et al. proposed a refined approach to conventional CSLA by eliminating redundant logic operations. Their method involved scheduling the carry select operation prior to the final sum calculation, which helped reduce the overall delay contributed by logic operations. In this refinement, logical optimization techniques were employed to enhance performance without excessively increasing complexity.

Carry Look-Ahead Adder (CLA): Carry look-ahead adders are another category of high-speed adders that aim to reduce carry propagation delay by predicting carry outputs in advance. While CLA significantly improves speed, it also requires more area—often more than twice that of an RCA. This area requirement can be a limiting factor in VLSI designs, especially when balancing speed with resource utilization.

Carry-Skip Adder (CSA): Carry-skip adders offer a hybrid approach by allowing the carry to skip over sections of the adder where it is not needed, effectively minimizing delay. The addition of incrementation and concatenation schemes to the fundamental carry-skip architecture has led to improved delay performance, though this can increase design complexity.

Hybrid Adder Structures: The need for efficient adder structures has led researchers to explore hybrid designs that combine various techniques to achieve optimal performance. A notable example is a modular hybrid adder designed specifically for high-throughput filter architectures. This design integrates algorithmic modifications using retiming and improvements in the performance of adders and multipliers to enhance overall efficiency.

Energy-Efficient Adder Designs: With the growing demand for low-power solutions in mobile and embedded systems, energy-efficient adder designs have gained prominence. Various researchers have focused on developing low-power full adders that retain competitive speed characteristics. For instance, a modified hybrid full adder based on Semi XOR-XNOR gates has shown promise in terms of speed and power consumption, although it lacks the comprehensive output generation capabilities of conventional XOR-XNOR gates.

In any VLSI design that involves arithmetic operations, the adder serves as a fundamental building block and often constitutes part of the critical path. Consequently, a wide array of architectures, extending beyond the traditional ripple carry adder (RCA), have been proposed to address this issue [33],[34]. These architectures primarily aim to reduce carry propagation delay. However, while the speed of the adder is enhanced, the area of the design typically increases. Notably, the carry look-ahead adder requires more than double the area of simpler designs. A delay- and energy-efficient modular hybrid adder, suitable for use in high-throughput filter architectures, is discussed in [25]. The modified hybrid full adder is a low-power design that performs well in terms of speed and power consumption. This design is based on Semi XOR-XNOR gates [36], but it lacks the capability to generate all possible outputs like conventional XOR-XNOR gates. Two levels of modifications are incorporated in this paper to enhance the performance of filter structures: (1) algorithmic modification using retiming and (2) improvements in the performance of adders and multipliers in the filter. The ripple carry adder (RCA) has low power consumption and requires less area compared to most other adder structures. However, the linear increase in delay with an increase in the adder's bit-width renders it unsuitable for high-speed applications. High-speed adder structures also include carry-skip adders, carry look-ahead adders (CLA), conditional adders, and their combinations [37–40].

The carry-select adder (CSLA), introduced by Bedrij, improves the speed of addition by generating multiple carries independently and producing the sum from the selected carry. Conventional CSLAs have lower propagation delay compared to RCAs, but the area overhead due to the presence of dual RCAs (one for input carry $cin=0$ and the other for $cin=1$) made them less attractive. While CSLAs are delay efficient compared to RCAs, the presence of dual RCAs for the carry chain makes them less attractive. One of the central challenges in adder design is managing the trade-offs between speed, area, and power consumption. While advanced structures like CLA and CSLA significantly reduce delay, they often result in increased area and power usage. Conversely, simpler structures like RCA may excel in area efficiency but suffer from unacceptable delay for high-speed applications. Designers must balance these factors according to the specific requirements of their applications. By regrouping filter taps and effectively retiming delays, improvements in critical path delay (CPD) and a reduction in the number of accumulation path adders were achieved. A fast adder using multiplexers and NOR gates is discussed in [41].

Adders are fundamental components needed in all types of filters and arithmetic units. The performance of a complex digital signal processing system can be significantly improved by incorporating a delay, area, and energy-efficient adder structure in place of conventional adder structures. Various adder structures have been explored by researchers to meet their design requirements and constraints [42],[43]. The ripple carry adder (RCA) is straightforward to design and requires relatively low area. However, the carry propagation throughout the entire adder stages results in large propagation delay (T_{pd}), making it unsuitable for high-speed applications. To minimize the propagation delay of adders, carry-select adders (CSLA) and carry look-ahead adders (CLA) were devised. Delay-efficient adder structures are critical in modern VLSI design, especially in applications requiring high-speed arithmetic operations such as digital signal processing and advanced computational tasks. While various architectures, including RCA, CLA, CSLA, and modified versions of these adders, have been developed to address the challenges associated with speed, area, and power consumption, each comes with its trade-offs. Continuous research and innovation in adder design are essential to meet the growing demands of high-performance computing. By understanding the strengths and weaknesses of each approach, designers can make informed decisions that align with their specific application requirements, ultimately leading to more efficient and effective VLSI systems.

As technology continues to evolve, the demand for faster, smaller, and more efficient adder structures will only increase. Future research may focus on: Advanced Logic Techniques: The development of novel logic techniques that can further reduce the area and power consumption of adders while maintaining or improving speed, Integration with Other Components: Exploring ways to integrate adder designs with other arithmetic units and DSP components to create holistic solutions that optimize performance across the board, Adaptation to Emerging Technologies: As VLSI technology progresses toward smaller nodes, adapting adder designs to leverage the unique characteristics of new fabrication processes will be essential and Machine Learning Approaches: The application of machine learning techniques in optimizing adder design and performance could lead to groundbreaking advancements in both speed and efficiency.

III.HIGH THROUGHPUT FIR FILTER DESIGN

The length of the critical path in a direct form FIR filter increases as the filter lengthens. On the other hand, the transposed FIR filter benefits from a shorter critical path since it involves only a multiplication and an addition operation. However, this advantage is offset by the transposed FIR filter's increased input capacitance and larger area compared to the direct form FIR filter. In the transposed FIR filter, the accumulation path word length is always greater than the input path word length when the signal is not rounded or truncated. This extended word length includes the input path word length, the coefficient word length, and additional guard bits needed to prevent overflow in the accumulation path. Consequently, the register bit-width in the accumulation path increases, leading to higher input capacitance and a greater area requirement than in the direct form implementation. Lee et al. introduced a strategy that combines the transposed form and the direct form FIR filters to potentially reduce latency at the expense of increased area [44]. Furthermore, Kei-Yong Khoo et al. proposed an efficient implementation of FIR direct form structures by subdividing them into sections that adhere to a cycle-time constraint [45]. The research in FIR filter design has made significant strides in optimizing performance, reducing complexity, and improving energy efficiency. Through innovative algorithms, hardware optimization techniques, and efficient implementation strategies, FIR filters have become more effective and versatile, meeting the growing demands of modern digital signal processing applications. These advancements not only enhance the capabilities of FIR filters but also pave the way for further innovations in the field. As the filter length increases, the critical path length of a direct form FIR filter also increases. Algorithmic transformation techniques, including pipelining and parallel processing, can be used to improve the input data processing rate or throughput. Pipelining improves throughput at the cost of increased latency[46].

This paper focuses on improving the throughput and energy efficiency of an FIR filter while minimizing latency and hardware complexity. The two levels of modifications incorporated in this paper to improve the performance of filter structures are (1) algorithmic modification using retiming and (2) improving the performance of adders and multipliers in the filter. The use of FIR filters simplifies the fixed-point implementation of DSP architectures and suits multi-rate applications. The critical path delay in a transposed FIR filter structure is the combined delay of an adder and a multiplier, expressed as:

$$t_{tf} = t_{tm} + t_{am}$$

Where t_{tm} is the time for a multiply operation, and t_{am} is the time for the highest bit-width addition in the accumulation path. From this equation, it is evident that reducing the bit-width of the accumulation path words can decrease t_{am} , thereby improving the filter's throughput and reducing register complexity. Additionally, t_{am} can be minimized by using delay-efficient adders instead of conventional ones. The principles of adaptive LMS filters and their applications are detailed in [47]. Retiming can enhance throughput without increasing latency. Therefore, a methodology that relies more on retiming than on pipelining and parallel processing is proposed for constructing a modified transposed FIR filter structure and an adaptive filter structure subsequently. In recent research, efforts have been made to identify the most energy-efficient systems for implementing FIR filters. Two main approaches for Multiply and Accumulate (MAC) operations are considered: the Multiple Constant Multiplication (MCM) system and the Distributed Arithmetic (DA) system. Simulation and synthesis tools were utilized to evaluate these systems, revealing that the DA system is highly effective in terms of memory and power consumption. DA uses fewer memory units, thereby simplifying the filter's design complexity, Ratnala Vinay[48]. In another study, Nithish Kumar V[49] and colleagues proposed a modified MAC unit-based FIR filter implementation that is both energy and space-efficient. They analyzed the performance of this FIR filter, created using both standard adders and modified carry select adders. Their Verilog HDL-designed 5-tap and 9-tap FIR filter architectures, implemented using SAED 90nm CMOS technology, demonstrated significant improvements. The Area Delay Product (ADP) for the 5-tap and 9-tap filters improved by 18.26% and 13.94%, respectively, over traditional methods. Similarly, the Power Delay Product (PDP) improved by 16.80% and 12.54%, respectively. To achieve the desired frequency response in FIR filters, window functions are utilized to balance the ripples in the passband with the sharpness of the transition band [50].

Digital filters are crucial in the field of Digital Signal Processing (DSP), with FIR filters being the most commonly used. FIR filters are extensively applied in image filtering, frequency modulation, precision arithmetic, and other areas. To optimize the design of these filters, various techniques are employed to enhance performance across different parameters, such as control parameter dependence and premature convergence. The advantages of these optimization techniques include simple design implementation, minimized error function, superior search capability, and fast convergence. FIR filters come in several types, including low pass, high pass, band pass, and band stop filters. A low pass filter allows only signals below a specified cutoff frequency to pass through, effectively eliminating high-frequency components. This makes low pass filters ideal for applications like audio signal processing, where they can limit the range of frequencies. High pass filters, in contrast, block frequencies below a certain threshold. An example application is to remove the 60Hz AC power "hum" noise from signals[51]. Bandpass filters are used in devices like cell phones, ensuring that only signals within a narrow, government-authorized frequency range are transmitted. Band stop filters block a specific range of frequencies while allowing low and high frequencies to pass, useful in various engineering applications. Recent advancements in computational algorithms have led to the development of FIR filter architectures that do not rely on multipliers, known as multiplier-less architectures. These architectures use power-of-two coefficients, which can be optimized using the Differential Evolution (DE) algorithm. Different DE mutation techniques were analyzed, leading to a new self-adaptive DE algorithm that improves the design process. Dempster and Macleod contributed significantly to reducing adder costs with their n-dimensional simplified adder graph (RAG-n) algorithm. This algorithm is divided into two parts: the first guarantees minimal adder costs by thoroughly synthesizing a set of coefficients, and the second uses heuristic algorithms to generate optimal multiplication costs from lookup tables, Dempster and Macleod[52]. In the same vein, Li proposed a method to reduce the complexity of fixed-point multipliers with fixed or programmable multiplicands. Their approach focuses on minimizing the number of adders required for a given multiplicand. This technique, which uses minimum number of shift-add operations (MNSAO), significantly increases the range and number of representable integers, reducing the mean approximation error compared to traditional CSD expressions under the same number of shift-add operations (Li) [53]. The concept of multiplier-less digital filters was further refined, taking into account pre-specified implementation costs calculated based on the total number of adders in the entire filter. The results indicated that the filters designed using these new methods significantly outperformed those designed with MILP programming and simulated annealing (SA), which limited the number of SPT terms for each coefficient. The research landscape for FIR filter design has seen substantial advancements in optimizing energy efficiency, area, and performance.

Approaches such as the DA system, modified MAC units, and various optimization algorithms have led to significant improvements in filter design. These methods have demonstrated their effectiveness in reducing complexity, power consumption, and area, while maintaining or enhancing the performance of FIR filters. As digital signal processing continues to evolve, these innovative techniques will likely play a crucial role in the development of more efficient and effective digital filters. In recent research, various methodologies and optimization techniques have been proposed to enhance the performance, reduce the complexity, and improve the energy efficiency of FIR filter designs. These advancements encompass both algorithmic improvements and hardware implementations, providing significant benefits over traditional methods. Another promising approach involved optimizing the conversion process to decrease the number of additions and subtractions for a given range of filter coefficient values and coefficient representation schemes. This was achieved by eliminating typical subexpressions in the coefficients' binary representation for a direct form FIR filter structure. For the transposed form, the reduction of adders was also considered. By integrating the CSE algorithm, the total number of addition and subtraction operations for direct structures was reduced by 35%, and the total number of subtraction operations for transposed structures was reduced by 38%. Overall, the average reduction in the total number of additions and subtractions was a factor of 2.2 compared to 1.43, as seen in prior references.

A novel approach involved in designing low-power FIR filters through parallel or block processing with hardware replication. By doubling the number of adder elements, they significantly reduced the number of multiplier elements. Since the area required for multiplier elements is substantially greater than that for adder components, this approach effectively reduced hardware costs and power consumption. Subsequently, an adjacent coefficient sharing-based substructure sharing approach was employed to further decrease hardware expenses in parallel FIR filters, coupled with the maximal absolute difference quantization technique. Their solution demonstrated a 45% reduction in hardware costs compared to conventional parallel filtering methods. Additionally, a truncated MCM using pattern modification technology (PMT) was developed for FIR filter implementation [54]. This algorithm truncates node adders generated by various MCM algorithms in DAG, ensuring uniform weight for every two inputs to the same node. The PMT algorithm reduced area costs by 35% relative to non-truncated MCM algorithms without increasing quantization error, making it suitable for delay and power-efficient applications. They explored different techniques and methodologies aimed at improving the hardware efficiency of FIR filters. Efficient retiming of large circuits was discussed in [55], while the application of retiming techniques to sequential circuits was examined in [56]. Jiang and Brayton [57] synthesized logic designs from a complexity perspective, providing insights into optimizing FIR filter structures. Yagain and Vijaya [58] applied retiming to multiplier-less multiple constant multiplication (MCM) algorithms, demonstrating its potential to enhance filter performance. Meher [59] improved the critical path delay (CPD) of conventional direct form filters by applying cut-set and flexible retiming techniques, although this resulted in increased hardware complexity. Thakral et al. [60] applied unfolding transformation to FIR filters using carry increment adders (CIA) and Vedic multipliers, showcasing a significant improvement in filter performance. Swati and Himanshu [61] developed FIR filters using Wallace multipliers, achieving notable gains in efficiency and performance. Rai et al. [62] utilized carry increment adders and Vedic multipliers for FIR filter implementation, resulting in improvements in CPD and energy efficiency at the cost of increased area compared to conventional direct form FIR filters. Meher [63] explored various pipelining possibilities in DSP circuits, contributing to the advancement of high-performance digital filters. Pramod and Shahana [64] developed a high-throughput adaptive filter architecture using a modified transposed form FIR filter, demonstrating significant performance improvements. They also explored the application of retiming transformation techniques for developing low-power VLSI architectures [65]. Furthermore, Pramod and Shahana [66] developed two versions of high-throughput FIR filters using retiming and modified CSLA-based adders, showcasing substantial advancements in filter design and implementation. Krishnamurthy et al. [67] designed FIR filters using novel pipelined bypass multipliers and carry look-ahead adders, achieving improvements in delay compared to conventional direct form FIR filters at the cost of increased hardware complexity.

Pramod and Shahana [68] developed modified transpose form FIR filters for their adaptive filter architecture. By regrouping filter taps and effectively retiming delays, they achieved improvements in critical path delay (CPD) and reductions in the number of accumulation path adders. Pramod Kumar Meher and Abbes Amira [69], in their paper entitled "FPGA realization of FIR filters by efficient and flexible systolization," presented the realization of 8-tap and 16-tap Digital FIR filters by systolic decomposition of distributed arithmetic (DA). Implemented on Xilinx Virtex-E XCV2000E FPGA using a hybrid combination of Handel-C and parameterizable VHDL cores, their approach analyzed the performance based on the maximum operating frequency. The implementation was found to have less area delay complexity, with the 8-tap FIR filter achieving a maximum operating frequency of 74.025 MHz and the 16-tap FIR filter achieving 67.222 MHz. Rakhi Thakur and Kavita Khare [70], in their paper entitled "High Speed FPGA implementation of FIR filter for DSP Applications," presented an approach to the high-speed FPGA implementation of FIR filters. FPGAs offer higher sampling rates and lower costs than ASICs.

Their paper described an approach to implementing digital filters based on FPGAs, coded in VHDL. The performance analysis was based on parameters such as the minimum period of 4.255 ns and a maximum frequency of 235.026 MHz. The results required low area and total memory usage of 147,920 kilobytes.

The advancements in FIR filter design and implementation have led to significant improvements in performance, energy efficiency, and hardware optimization. Through the application of algorithmic modifications, such as retiming and pipelining, and the use of innovative hardware components, FIR filters have become more capable of meeting the demands of modern digital signal processing applications. These developments not only enhance the functionality and efficiency of FIR filters but also contribute to the ongoing evolution of digital filter design.

IV. CONCLUSIONS

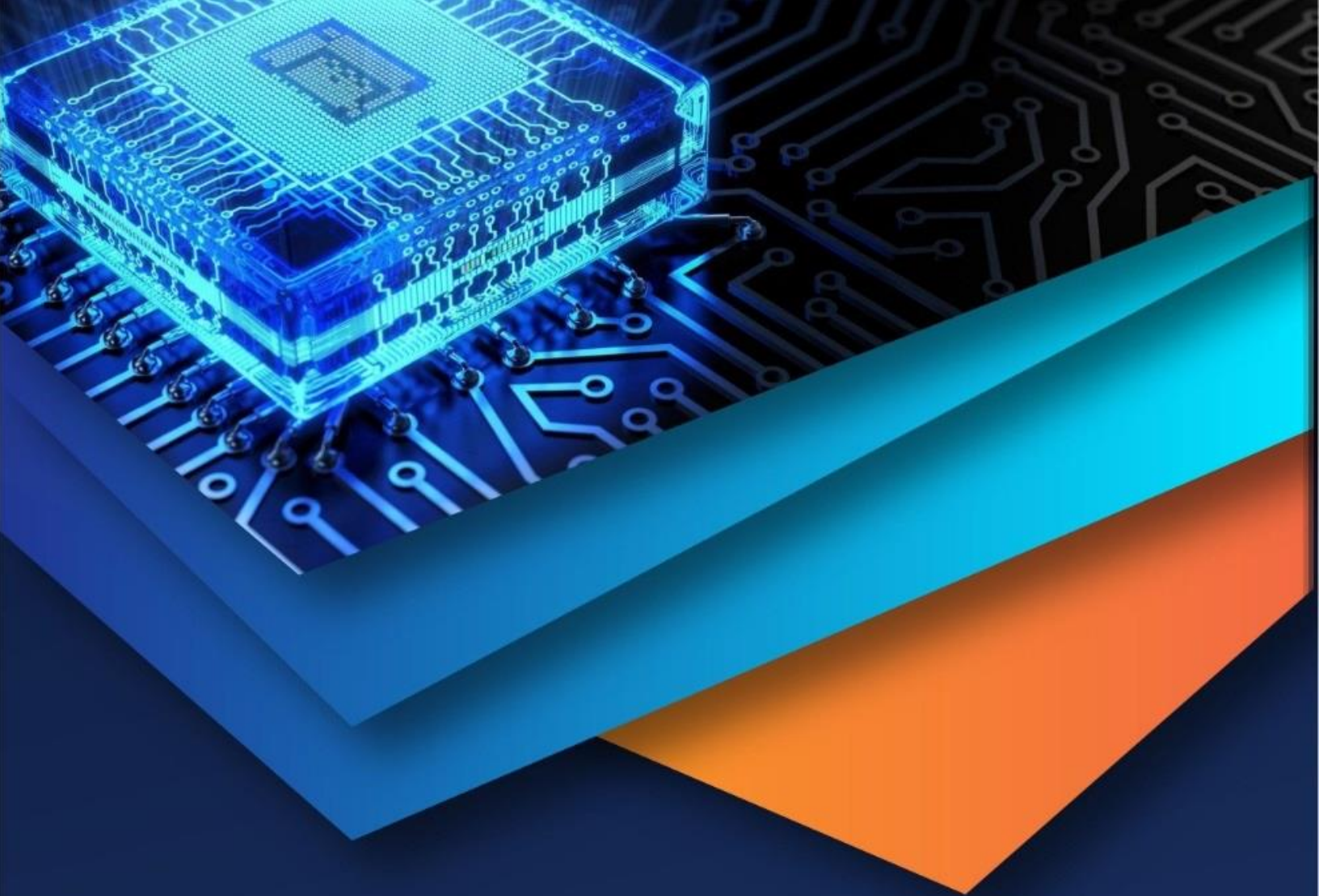
The evolution of FIR filter design has been pivotal in advancing the field of Digital Signal Processing (DSP), addressing the critical need for high-performance and efficient digital filters. FIR filters, known for their stability, linear phase response, and computational efficiency, have become essential in a wide array of applications, including image processing and frequency modulation. Their robustness against finite precision arithmetic errors and compatibility with Fast Fourier Transformation (FFT) techniques further enhance their appeal over Infinite Impulse Response (IIR) filters in many scenarios. The design of digital filters has seen significant advancements through various optimization techniques and architectures. Transposed FIR filters, adaptive filters, and distributed arithmetic (DA) have been instrumental in improving filter performance. The development of high-speed adder structures, such as carry-lookahead adders, has played a crucial role in enhancing the efficiency and effectiveness of digital filters. Recent research has made substantial strides in optimizing FIR filter architectures via algorithmic and hardware modifications, including retiming, pipelining, and parallel processing. Innovations like multiplier-less architectures, power-of-two coefficients, and advanced adder designs have significantly improved filter performance by reducing complexity and enhancing energy efficiency. Techniques such as Distributed Arithmetic (DA) and Multiple Constant Multiplication (MCM) have been particularly effective in minimizing memory usage and power consumption, making them ideal for high-speed, low-power applications. Ongoing exploration and refinement of FIR filter design are vital for the progression of DSP. These advancements are key to developing high-performance, efficient digital filter implementations that meet the increasing demands of modern technology. As research continues to uncover new optimization strategies, the resulting innovations will drive further progress across various applications, cementing the importance of FIR filters in the future of digital signal processing.

REFERENCES

- [1] Jiang, Lei, Haijian Zhang, Shuai Cheng, HengweiLv, and Pandong Li. "An Overview of FIR Filter Design in Future Multicarrier Communication Systems." *Electronics* 9, no. 4 (2020): 599.
- [2] Parhami, B. *Computer arithmetic*. Oxford university press, 1999.
- [3] Wanhammar, L. *DSP integrated circuits*. Elsevier, 1999.
- [4] N. West and K. Eshraghian, *Principles of CMOS VLSI Design: A System Perspective*, 2nd ed., Addison Wesley, NY, 1992. pp. 515-517, 1993. 121
- [5] J. M. Rabay, *Digital Integrated Circuits, A Design perspective*. Englewood Cliffs, NJ: Prentice-hall. 1996, pp. 392-393.
- [6] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*. New York: Wiley, 1999.
- [7] K.K. Parhi, *VLSI Digital Signal Processing Systems Design and implementation*. Interscience, 2007.
- [8] Rabaey, J.M., Chandrakasan, A., Nikolic, B.: 'Digital integrated circuits: a design perspective' (Pearson Education, India, 2017, 2nd edn.)
- [9] John G. Prokis. Dimitris G. Manolakis. *Digital Signal Processing*. Fourth Edition. 2007
- [10] Mitra, S. K. (1998). *Digital Signal Processing: A Computer-Based Approach*. New York, NY: McGraw-Hill.
- [11] B.N Mohan Kumar, H.G Rangaraju, "Array Multiplier and CIA based FIR Filter for DSP applications," 2021 Special Issue of First International Conference on Engineering, Science, and Technology (ICEST), 10.47392/irjash.2021.020
- [12] A. Bisoyi, M. Baral and M. K. Senapati, "Comparison of a 32-bit Vedic multiplier with a conventional binary multiplier," 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, 2014, pp. 1757-1760, doi: 10.1109/ICACCCT.2014.7019410.
- [13] K. D. Rao, P. V. Muralikrishna and C. Gangadhar, "FPGA Implementation of 32 Bit Complex Floating Point Multiplier Using Vedic Real Multipliers with Minimum Path Delay," 2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), 2018, pp. 1-6, doi: 10.1109/UPCON.2018.8597031.
- [14] K. Dey and S. Chattopadhyay, "Design of high performance 8 bit binary multiplier using vedic multiplication algorithm with 16 nm technology," 2017 1st International Conference on Electronics, Materials Engineering and Nano Technology (IEMENTech), 2017, 10.1109/IEMENTECH.2017.8076956. pp. 1-5, doi: 10.1109/IEMENTECH.2017.8076956.
- [15] G. C. Ram, Y. R. Lakshman, D. S. Rani and K. B. Sindhuri, "Area efficient modified vedic multiplier," 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), 10.1109/ICCPCT.2016.7530294. 2016, pp. 1-5, doi: 10.1109/ICCPCT.2016.7530294.
- [16] K. D. Rao, C. Gangadhar and P. K. Korrai, "FPGA implementation of complex multiplier using minimum delay Vedic real multiplier architecture," 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON), 2016, pp. 580-584, doi: 10.1109/UPCON.2016.7894719.

- [17] Jiang, H., Han, J., Qiao, F., et al.: 'Approximate radix-8 booth multipliers for low-power and high-performance operation', Trans. Comput., 2016, 65, (8), pp. 2638–2644, doi: 10.1109/TC.2015.2493547
- [18] Qian, L., Wang, C., Liu, W., et al.: 'Design and evaluation of an approximate wallace-booth multiplier'. IEEE Int. Symp. Circuits and Systems (ISCAS), Montreal, QC, Canada, May 2016, pp. 1974–1977.
- [19] Ravindra P Rajput, M. N Shanmukha Swamy, "High speed modified Booth Encoder Multiplier for Signed and Unsigned Numbers," 14th International Conference on Modelling and Simulation, Pages: 649 – 654, March 28-30, 2012.
- [20] Uvaraj Subramanian, Srinivasan Alavandar, "Low-Power and high speed computation using hybridized Multiplier," Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT) IEEE 31661, July 4-6, 2013.
- [21] Himanshu Bansal, K. G. Sharma, Tripti Sharma, "Wallace tree multiplier designs: a performance comparison review", Innovative Systems Design and Engineering ISSN 2222-1727 (Paper), ISSN 2222-2871 (Online) Volume.5, Number.5, 2014.
- [22] Damarla Paradasaradhi, M. Prashanthi, and N Vivek, "Modified Wallace tree multiplier using efficient Square Root Carry Select Adder," International Conference on Green Computing Communication and Electrical Engineering (ICGCCCE), March 2014.
- [23] Nishat Bano, "VLSI Design of Low Power Booth Multiplier," International Journal of Scientific & Engineering Research, ISSN 2229-5518, Volume 3, Issue 2, February 2012.
- [24] Kavita, Jasbir Kaur, "Design and Implementation of an Efficient Modified Booth Multiplier using VHDL," International Conference on Emerging Trend in Engineering and Management (ICETEM), e-ISSN:2231-0347, Print-ISSN: 2231-2013, pp. 78 81, Volume.3 (3), July 2013.
- [25] Sukhmeet Kaur, Suman, and Manpreet Singh Manna, "Implementation of Modified Booth Algorithm (radix-4) and its Comparison with Booth Algorithm (radix-2)," Advance in Electronic and Electric Engineering, ISSN 2231-1297, Volume 3, Number 6, pp. 683-690, 2013.
- [26] Chang, C. H., Gu, J. and Zhang, M. A review of 0.18- μm full adder performances for tree structured arithmetic circuits. IEEE Transactions on very large scale integration (VLSI) systems 13 (6) (2005) 686 695.
- [27] Zlatanovici, R., Kao, S. and Nikolic, B. Energy–delay optimization of 64-bit carry-lookahead adders with a 240 ps 90 nm CMOS design example.
- [28] Ramkumar, B., and Kittur, H. M. Low-power and area-efficient carry select adder. IEEE transactions on very large scale integration (VLSI) systems 20 (2012) 371-375.
- [29] Mohanty, B. K., and Patel, S. K. Area–delay–power efficient carry-select adder. IEEE transactions on circuits and systems II: express briefs 61 (6) (2014) 418-422.
- [30] Bahadori, M., Kamal, M., Afzali-Kusha, A. and Pedram, M. High-speed and energy-efficient carry skip adder operating under a wide range of supply voltage levels. IEEE Transactions on very large scale integration (VLSI) systems 24 (2) (2016) 421-433.
- [31] Chang, T.-Y. and Hsiao, M.-J., "Carry-select adder using single ripple-carry adder". Electronics Letters. Volume: 34, Issue: 22, 29 Oct. 1998, Pages: 2101 - 2103
- [32] Youngjoon Kim and Lee-Sup Kim. "A low power carry select adder with reduced area". Circuits and systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on, Volume: 4, 2001. Pages: 218–221.
- [33] P. Pramod and T.K. Shahana, "Delay and Energy Efficient Modular Hybrid Adder for Signal Processor Architectures," IETE Journal of Research, Print ISSN: 0377-2063 Online ISSN: 0974-780X, 2 June 2019 <https://doi.org/10.1080/03772063.2019.1627917>
- [34] Vahid Foroutan, Mohammad Reza Taheri, Keivan Navi, Arash Azizi Mazreah, "Design of two Low-Power full adder cells using GDT structure and hybrid CMOS logic style" Integration the VLSI Journal (Elsevier) (2013) .
- [35] Doran, R.W.: 'Variants of an improved carry look ahead adder', IEEE Trans. Comput., 1988, 37, (9), pp. 1110–1113
- [36] Lehman, M., Burla, N.: 'Skip techniques for high-speed carry propagation in binary arithmetic units', IRE Trans. Electron. Comput., 1961, 10, (4), pp. 691–698.
- [37] Han, T., Carlson, D.A.: 'Fast area-efficient VLSI adders'. Proc. 8th IEEE Symp. Computer Arithmetic, Como, Italy, May 1987, pp. 49–56
- [38] Knowles, S.: 'A family of adders'. Proc. 14th IEEE Symp. Computer Arithmetic, Adelaide, Australia, 1999, pp. 14–16
- [39] Ceiang, T.Y., Hsiao, M.J.: 'Carry-select adder using single ripple carry adder', Electron. Lett., 1998, 34, (22), pp. 2101–2103
- [40] Kim, Y., Kim, L.S.: '64-bit carry-select adder with reduced area', Electron. Lett., 2001, 37, (10), pp. 614–615
- [41] Ardeshtir, G., Tavakolaee, H., Baleghi, Y.: 'Fast mux-based adder with low delay and low PDP', J. AI Data Min., 2019, 7, (3), pp. 385–392
- [42] R. P. Brent, and H. T. Kung, "A regular layout for parallel adders," IEEE Trans. Comp., Vol. C-31, no. 3, pp. 260–4, 1982.
- [43] Y. He, and C.-H. Chang, "A power-delay efficient hybrid carry lookahead/carry-select based redundant binary to two's complement converter," IEEE Trans. Circuits Syst.-I Reg. Pap., Vol. 55, no. 1, pp. 336–46, Feb. 2008.
- [44] Lee, H. R., Jen, C. W. and Liu, C. M. A new hardware-efficient architecture for programmable FIR filters. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing 43 (9) (1996) 637-644.
- [45] Khoo, K. Y., Yu, Z. and Willson, A. N. Design of optimal hybrid form FIR filter. In ISCAS 2001. The 2001 IEEE International Symposium on Circuits and Systems 2 (2001) 621-624.
- [46] Meher, P. K. and Park, S. Y. Area-delay-power efficient fixed-point LMS adaptive filter with low adaptation-delay. IEEE Transactions on Very Large Scale Integration (VLSI) Systems 22 (2) (2014) 362-371.
- [47] Safarian, C., Ogunfunmi, T., Kozacky, W. J. and Mohanty, B. K. FPGA implementation of LMS-based FIR adaptive filter for real time digital signal processing applications. IEEE International Conference on Digital Signal Processing (DSP), 2015, 1251-1255.
- [48] Leiserson, C. E., Rose, F. M. and Saxe, J. B. Optimizing synchronous circuitry by retiming (preliminary version). In Third Caltech conference on very large scale integration, 1983, 87-116.
- [49] R. Vinay, T. S. V. S. Vijayakumar, L. M. Saini and B. Singh, "Power efficient FIR filter Architecture using Distributed Arithmetic Algorithm," 2020 First IEEE International Conference on Measurement, Instrumentation, Control and Automation (ICMICA), 2020, 10.1109/ICMICA48462.2020.9242720. pp. 1-4, doi: 10.1109/ICMICA48462.2020.9242720.
- [50] V. N. Kumar, K. R. Nalluri and G. Lakshminarayanan, "Design of area and power efficient digital FIR filter using modified MAC unit," 2015 2nd International Conference on Electronics and Communication Systems (ICECS), 2015, pp. 884-887, doi: 10.1109/ECS.2015.7125041.

- [51] Chandra, Abhijit, and Sudipta Chattopadhyay. "Design of hardware efficient FIR filter: a review of the state-of-the-art approaches." *Engineering Science and Technology, an International Journal* 19, no. 1 (2016): 212-226.
- [52] D. L. Maskell, "Design of Efficient Multiplierless FIR filters," *IET Circuits Device System*, vol.1, no. 2, pp. 175-180, May 2007.
- [53] Dempster, Andrew G., and Malcolm D. Macleod. "Use of minimum-adder multiplier blocks in FIR digital filters." *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 42.9 (1995): 569-577.
- [54] Negi, Vivek. *Analysis and Design of Power Efficient Multiplierless Digital FIR Filter using Radix-2r Algorithm*. Diss. 2018.
- [55] N. Maheshwari and S. Sapatnekar, "Efficient retiming of large circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 6, no. 1, pp. 74-83, 1998.
- [56] C.E. Leiserson and J.B. Saxe, "Retiming synchronous circuitry," *Algorithmica*, no. 6, pp. 5-35, 1991.
- [57] J.R. Jiang and R.K. Brayton, "Retiming and resynthesis: a complexity perspective," *IEEE Trans. Computer-Aided Design of Integr. Circuits Syst.*, vol. 25, no. 12, pp. 2674-86, 2006.
- [58] D. Yagain and K.A. Vijaya, "Design of synthesizable, retimed digital filters using FPGA based path solvers with MCM approach: comparison and CAD tool," *VLSI Design*, Hindawi Pub. Corp., Article ID 280701.
- [59] P.K. Meher, "On efficient retiming of fixed-point circuits," *IEEE Trans. VLSI Syst.* vol. 4 no. 4, 2016.
- [60] S. Thakral, D. Goswami and R.S. Sharma, "Design and implementation of a high speed digital FIR filter using unfolding," *Proc. IEEE 7th Int. Con.*, Bikaner, India, Nov. 2016.
- [61] C. Swati and J. Himanshu, "FIR filter designing using wallace multiplier," *International Journal of Engineering and Technical Research*, vol. 3, no. 6, pp. 276-78, 2015.
- [62] S.N. Rai, B.S.P. Shree and Y.P. Meghana, "Design and implementation of 16 tap FIR filter for DSP applications," *Proc. Sec. Int. Conf. Adv. in Electron. Comp. and Comm.*, Bengaluru, India, pp. 9-10, feb. 2018.
- [63] P.K. Meher, "Seamless pipelining of DSP circuits," *J. Cir.Syst.SignalProcess.*, vol.35, o.4, pp. 1147-62, 2016.
- [64] P. Pramod and T.K. Shahana, "High throughput adaptive filter architecture using modified transpose form FIR filters," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 10, no.15, pp. 68-82, 2018.
- [65] S. Jalaja and V.A.M. Prakash, "Different retiming transformation technique to design optimized low power VLSI architecture," *AIMS Electronics and Electrical Engineering*, vol. 2, no. 4, pp. 117-130, 2018.
- [66] P. Pramod and T.K. Shahana, "High throughput FIR filter architectures using retiming and modified CSLA based adders", *IET Circuits, Devices and Systems*, Print ISSN 1751-858X, Online ISSN 1751-8598, 30 May 2019, <https://doi.org/10.1049/iet-cds.2019.0130>
- [67] Krishnamurthy, S., Kannan, R., Azwan, E., et al.: 'Design of FIR filter using novel pipelined bypass multiplier'. *Proc. Third Int. Symp. Robotics and Manufacturing Automation*, Universiti Putra Malaysia, Greater Kuala Lumpur Area, Malaysia, September 2017, pp. 19-21
- [68] Pramod, P., Shahana, T.K.: 'High throughput adaptive filter architecture using modified transpose form FIR filters', *J. Adv. Res. Dyn. Control Syst.*, 2018, 10, (15), pp. 68-82.
- [69] Pramod Kumar Meher, Abbas Amira, "FPGA realization of FIR filters by efficient and flexible systolization using Distributed Arithmetic," *IEEE Transactions On Signal Processing*.
- [70] Rakhi Thakur and kavita khare, "High Speed FPGA implementation of FIR filter for DSP Applications," *International Journal of Modeling and Optimization* , vol.3, No 1, February 2013.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)