https://doi.org/10.48047/AFJBS.6.Si4.2024.2787-2796

African Journal of Biological Sciences

Journal homepage: http://www.afjbs.com

Research Paper Open Access

Design and Implementation of Low Power Efficient Fir Filter Architectures Using Approximate Adders and Multipliers

Raningaram Thanuja Singh[1], Dr. K.Venkatachalam[2], Dr. K.Dhanunjaya[3], Eduru Mercy[4]

[1,4]M.Tech Scholars, Department of Electronics and Communication Engineering, Audisankara College of Engineering & Technology, Gudur, Andhra Pradesh, India.

[2,3]Professors, Department of Electronics and Communication Engineering, Audisankara College of Engineering & Technology, Gudur, Andhra Pradesh, India.

Article History
Volume 6, Issue Si4, 2024
Received: 01 June 2024
Accepted: 30 June 2024
Doi:
10.48047/AFJBS.6.Si4.2024.2787-2796

Abstract: Digital filters play a significant role in many signal processing applications. Every communication system and DSP (Digital Signal Processing) must have a Finite Impulse Response (FIR) filter. Two important components of filter architecture are the multiplier and adder. Digital circuits can be designed with a number of adders and multipliers, but an effective adder and multiplier design is required to create an efficient filter. While developing the digital Very Large Scale Integration (VLSI) circuits, an adder is one of the fundamental computational circuit. The concept behind Approximate Adders (AAs) is to improve the design metrics of the adders. The field of high-speed error-tolerant circuits with approximation computation has grown significantly due to the development of high-speed multimedia applications. Although these programs perform exceptionally well, accuracy is compromised. Their techniques additionally minimize system architecture complexity, delay, and power consumption. A proposal to improve the design metrics of an adder is called an approximate adder (AA). Multipliers are essential because they speed up the execution of intricate arithmetic procedures involving large numbers. The difficulty of today's world is that a multiplier allows to complete an operation in a single step quickly rather than to execute several addition or subtraction operations. The design and implementation of low power efficient FIR filter architectures using approximate adders and multipliers is presented in this work. A simple and fast approximation adder is proposed for approximate multiplier architectures. Area, delay, and power are used to evaluate the FIR filter's performance.

Keywords: Digital Signal Processing (DSP), Digital Filters, Finite Impulse Response (FIR) Filters, Adder and Multiplier, Very Large Scale Integration (VLSI)

1. INTRODUCTION

The demand for image processing systems is increasing quickly in recent times. With the latest technology, every industry where the creation and storing of image data is crucial is seeing an increase in the number of active users. Image processing is the basis for many significant

applications, including object detection, medical imaging, remote sensing, and etc. The key components of these image processing methods are digital filters. In digital signal processing, digital filters play a vital role [1].

A device or process used to remove unwanted components or features from a signal is called a filter. One type of signal processing utilized in the represented synthesis of an arithmetic circuit is filtering, which is used to evaluate the digital design at the logical level. A filter is characterized by the complete or partial suppression of certain signal components. Although the utilization of a field-programmable gate array based on signal processing, the existing approach is examined in order to increase each system's level of efficiency. For efficient digital computation, noiseless filter design presents additional challenges for signal processing [2].

As a preprocessing step, noise is widely removed from the detected signals of WSNs (Wireless Sensor Networks) using the FIR filter. As VLSI technology is on improving, FIR filters have to be implemented in real-time with a minimum of delay and hardware requirements. When presented with an input of finite length, like an impulse, the FIR filter's output is limited and eventually returns to zero. When using an Infinite impulse response (IIR) filter, the impulse response is unlimited. Applications for Digital signal processing (DSP) include speech recognition, SONAR (Sound Navigation and Ranging) RADAR (Radio Detection and Ranging), digital image processing, audio signal processing, telecommunications, and etc. Big data applications, real-time limitations, and low energy requirements are some of the challenges faced by DSP [3]. Applications involving digital signal processing frequently make use of FIR (Finite Impulse Response) filters. It is essentially made up of several delays, multipliers, and adders. Enhancing the subcomponents power efficiency and compactness will contribute to the filters reduction and low power consumption, which may result in cost savings and longer battery life for portable devices [4].

FIR indicates that this filter doesn't use feedback and that the impulses are finite. Phase is maintained linear to prevent noise distortions. The design of FIR is simpler than that of IIR. DSP processors utilize these types of FIR filters because of their high speed. Addition and multiplication are utilized at times in digital signal processing. FIR filters in digital signal processing define fewer bits are created with finite precision [5].

Nowadays, the area of VLSI design implementation for particular image processing applications is observe the development of approximation computations [6]. The area of high-speed error-tolerant circuits with approximation computation has expanded significantly due to the development of high-speed multimedia applications. These programs offer excellent performance since at the cost of decreased accuracy. Moreover, their approaches reduce power consumption, delay, and system architecture complexity [7]. The complex and energy-hungry blocks in the filter architectures are multipliers. Today's digital signal processing and many other applications depend heavily on multipliers [8]. In order to develop multipliers that are appropriate for a number of high speed, low power, and compact VLSI implementations, many researchers have tried and continue to try to design multipliers that offer high speed, low power consumption, regular layout, and so less area, or even a combination of these design targets in one multiplier. This is due to technological advancements [9].

Multipliers play an important role in today’s digital signal processing and various other applications. With advances in technology, many researchers have tried and are trying to design multipliers which offer either of the following design targets – high speed, low power consumption, regularity of layout and hence less area or even combination of them in one multiplier thus making them suitable for various high speed, low power and compact VLSI

implementation [10]. So optimization is commenced with the reduction in the number of multipliers. Approximation is suggested using half-adder, full-adder, and compressors. The complexity of the arithmetic units has been lowered, but the error value has also been taken into consideration. Systemic approximation improves accuracy, while lower logic architecture of approximation arithmetic units reduces power and area consumption [11].

There are several adders and multipliers in the FIR filter design, and in order for the filter output to reflect, a delayed component is required. The FIR filter does not compute any summation or multiplication error rounding. A large area size required for multiplier adders and delayed elements, depending on the filter order is one of the main drawbacks of the FIR filter architecture, along with its high propagation delay [12]. One of the main types of filters used in digital signal processing is the FIR filter. FIR filters are called finite since they don't have any data. Digital filters with Finite impulse response (FIR) can be used in a number of digital signal processing applications, such as Software-defined radio (SDR), loud speaker equalization, adaptive noise reduction, echo cancellation, and speech processing, in addition to other applications for communication [13]. Large-order FIR filters are needed for many of these applications in order to achieve the stringent frequency requirements. These filters frequently have to handle high sample rates in order to allow high-speed digital transmission [14]. However, the number of additions and multiplications needed for each filter output is increases in a linear way as the filter order increases. Real-time implementation of a big order FIR filter in a resource-constrained setting is a difficult issue since the FIR filter method does not allow for redundant computation.

To address these difficulties, they discuss the design and implementation of low-power FIR filter architectures that utilize approximation adders and multipliers. The remaining content is arranged as follows: The Literature survey is explained in Section 2. The section 3 presents design and implementation of low power efficient FIR filter architectures using approximate adders and multipliers. The analysis of the results is evaluated in section 4. In section 5, the conclusion is provided.

2. LITERATURE SURVEY

The 4:2 Compressor was designed for the Parallel Distributed Arithmetic (PDA) FIR Filter system. This work presents a new PDA FIR channel design was use of the 4:2 compressors that Xilinx FPGAs can effectively map. Considering comparison with the most advanced PDA FIR channel, this suggested FIR models achieve a 17.5% reduction in asset utilization and a 20.7% change in execution overall. In addition, when compared to PDA FIR, there has been an overall 57.9% decline in asset utilization and a 23.0% change in execution. A different 4:2 compressor design is suggested in light of several inward conditions that have changed. These Compressors can be widely used as building squares of multipliers [15].

A FIR filter structure was designed utilizing a Wallace tree multiplier and high adders. In order to design 8x8 bit multipliers, 4:2, 5:2, and 6:2 compressors are used. The suggested 4:2, 5:2, and 6:2 compressor designs with various logic styles are contrasted with the conventional Wallace tree design 14 transistor adder cells. The number of transistors and power consumption were significantly decreased in the suggested manner. The Wallace tree multiplier's area and speed are improved by up to 24% as a result of this compressor. The results show that, with regard to delay, the suggested architecture is more effective than the current one. For high-speed applications, multiplying values larger than 16 bits can be an appropriate match for this method [16].

The Implementation of Efficient FIR Filter was designed using Approximate Compressors for Multiplication. Approximate compressors were used for multiplication. This article proposes a number of new design strategies for approximately 4-2 and 5-2 compressors, with the goal of reducing the number of partial product steps involved in multiplication. Two novel approximate 5-2 compressors have been proposed for two approximate 8 × 8 Dadda multiplication designs, while three novel approximate 4-2 compressors have been proposed for three approximate 8 × 8 Dadda multiplier designs. According to the synthesis's results, in contrast to the current approximations, the suggested designs made significant improvements in accuracy along with reductions in power and delay [17].

A 5-stage FIR filter was designed with a modified Montgomery multiplier that has a high-performance, low-power architecture. All of the design simulation is done with 45 nm technology was CMOS (Complementary Metal Oxide and Semiconductor) and PTL (Pass-transistor Logic). The results show an increase in the multiplier's area efficiency of 65% and power reduction of approximately 68% when compared to the conventional approach. The multiplier's logic gates were designed using CMOS and pass transistor logic, which reduced the amount of transistors and produced a low-power design [18].

The Multiplierless Multiple-Stage Cascaded FIR Filter Design is explained. A new algorithm for designing multiplier-less multiple-stage cascaded FIR filters is presented in this study. The recommended algorithm's phase count is automatically computed, in comparison to traditional algorithms, where there are often two set phases. In terms of hardware cost and design time saved, the design examples demonstrate that the existing algorithm performs significantly better than the best existing approach [19].

Parallel Prefix Adder (PPA) and a dadda multiplier was designed to develop a programmable FIR filter. Here, PPA is used for addition, and the Dadda multiplier is utilized for multiplication. As a result, this effort contributes to the development of an effective FIR filter that has numerous DSP applications. Dadda multiplier takes up less space since it requires fewer gates. The process is executed in parallel, therefore the PPA benefits from faster computing. This is a programmable FIR filter since the user can select the type of filters (high pass, low pass, band pass, or band stop) based on the purpose of the filter in addition to providing the necessary number of coefficients [20].

The design of less effective length sparse FIR filters were presented. To produce FIR filters with sparse linear phase and shorter effective lengths, an exchange algorithm is proposed. For sparsity maximization, it might be helpful to use an iterative 0-1 exchange technique with suitable direction control to get the minimax approximation error is nearer to the intended upper bound of error. In the suggested approach, optimizing the effective length is given less importance than sparsity. According on simulation data, the suggested approach is generally better than the current algorithms in terms of effective length and/or sparsity [21].

A synthesizing approach for tap delays and accumulations that is cost-aware and used to construct FIR filters that are efficient in terms of area and power. In order to solve this problem of integer programming at quadratic computational cost, an effective Genetic Algorithm is developed that decreases the search space is to identify an optimal solution that satisfies the frequency response parameters. Results from twelve benchmark filter specifications, using application-specific integrated circuit logic synthesis and field programmable gate arrays, when compared to solutions produced by current design methods, this results generated by this suggested technique showed a reduction of up to 26.8% and 27.5% in average area and power consumption [22].

Design and implement a FIR filter utilizing adders and multipliers with high speed and low power. The Carry Look Ahead (CLA) adder and the Radix-4 modified Booth Multiplier are used to build and execute a high speed, low power FIR filter. When multiplying filter inputs and coefficients, the Booth multiplier shortens the accumulation computation time. The typical ripple carry adder, which is used to add the FIR filter, has a critical path delay that can be decreased with the use of CLA. Using a Booth multiplier and CLA, the 8-tap direct form FIR filter is constructed, simulated, and synthesized. Every block's utilization summary with regard to the target FPGA is also provided [23].

Design and implementation of a FIR filter structure utilizing Wallace Tree Multiplier and High Adders. This project uses 4:2, 5:2, and 6:2 compressors to produce 8x8 bit multipliers. The suggested 4:2, 5:2, and 6:2 compressor designs with various logic styles are contrasted with the conventional Wallace tree design 14 transistor adder cell. The number of transistors and power consumption were significantly decreased in the suggested manner. The results show that in terms of delay. The suggested architecture is more effective than the current one. In order to develop a high performance multiplier in VLSI applications, the power of the proposed multiplier might be examined [24].

Design and implementation of hybrid FIR filters was presented which uses fast adders and Vedic multipliers. In this work, they implement high-performance FIR filters by designing and analyzing several multipliers, adders. The sixteen Sutras that make up the Vedic Mathematics are a set of techniques for quick mental calculations. With a 1.5% delay reduction and a 10.2% power reduction, this method improves speed over previous methods [25].

A new design model was described for designing multi-dimensional FIR filters based on desired frequency response representation using a gabor system which is generated by the Gaussian function. The desired frequency filter frequency response was obtained from the specifications of standard design. This approach results stop-band attenuation and provides control over the critical frequencies which turn being close to the obtainable with Parks-McClellan model. This improved performance can be better option of design of 2-D filters [26].

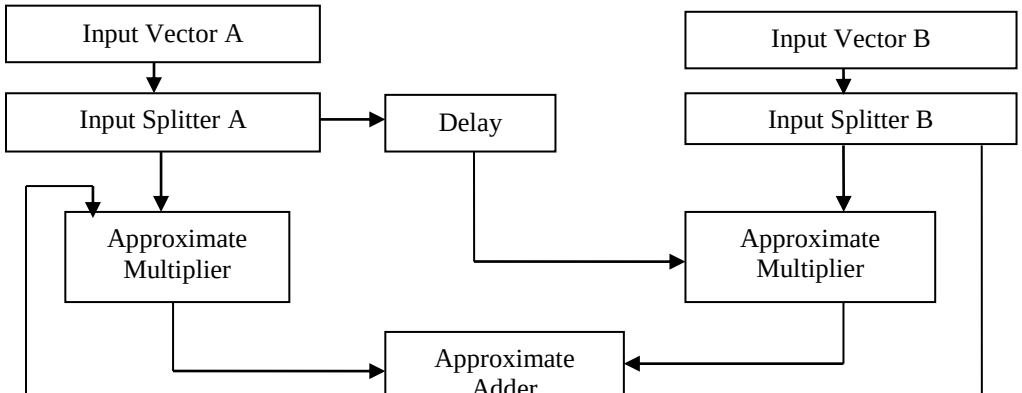
3. DESIGN AND IMPLEMENTATION OF LOW POWER EFFICIENT FIR FILTER ARCHITECTURES

In this section, design and implementation of low power efficient FIR filter architectures using approximate adders and multipliers is presented. The provided FIR filter's block diagram is shown in Figure 1. The linear-phase feature and stability of Finite impulse response (FIR) filters make them popular in Digital signal processing (DSP) applications.

Three fundamental components are needed to create a FIR filter. They are signal delay, addition, and multiplication. The expression for a FIR filter is expressed in equation (1) as follows:

$$y[n] = \sum_{k=0}^{N-1} b_k x(n-k) \qquad (1)$$

If the set of filter coefficients is identified by b_k, the output signal is represented by y [n], and N represents the filter order. Taps or tapped delay lines are information used to describe x [n - k] times when x[n] is the applied input signals. Provide a fundamental FIR filter multiplier structure in the standard design. It creates partial products in order to carry out multiplication.



Approximate multipliers and adders are utilized in this analysis to enhance the FIR filter's performance. Designing a FIR filter is requires a set of coefficients and a sample delay line. To put the filter into application: In the delay line, enter the input sample. After multiplying each sample in the delay line by the corresponding coefficient, added the result that is obtained. When implementing a FIR filter using filter(), they set the vectors expressed as equation (2)

$A = [1] and\ B = [B_0, B_1, \ldots\ldots, B_M]$ (2)

One sample of the signal delay is provided by the unit delay. After one sample clock cycle, a sample value is kept in a memory slot and available as an input to the subsequent processing step. Over the frequency range, Phase in linear form, group delays in FIR filters are uniform and equal.

An order n linear phase FIR filter simply delays the filtered signal by n/2 time steps, and the magnitude response of the filter determines the scale of its Fourier transform. The group delay is n/2. Phase distortion is eliminated by this feature, which maintains the shape of signals within the pass band. An approximate multiplier takes the place of conventional multipliers. For energy-efficient computing in applications where there is an inherent tolerance for error, approximate multipliers are heavily recommended. A multiplication process consists of two steps: the development of partial products and their addition. Therefore, the speed at which they can combine the speed of the approximate multiplier is mainly determined by the partial products after they have been formed. The speed at which partial products are generated, can be achieved by producing fewer of them. In order to speed up the process of adding among incomplete products, they need fast adder designs. The proposed approximation multiplier makes use of error signals are increase accuracy through error correction and partial product accumulation using a simple tree of approximation adders. In order to create accurate but energy-efficient DSP cores, this model is revolutionary, this approximation design has developed throughout time.

Error vector and approximate is produced by the proposed approximate adder, which prevents "carry propagation". The Partial product (PP) is accumulated through an adder tree. The result is produced with more accuracy when using the trees to made up for output errors. Since an approximation adder is simpler than a regular adder and produces a much lower error signal in terms of Critical-Path Delay (CPD), it is an excellent option for creating an adder tree to accumulate values. The created adder breaks the CPC (Critical Path Chain) is to calculate data in parallel while generating an approximate sum and error vector.

The output y[n] is expressed in equation (3) and (4) as

$y[n] = v_1[n] + v_2[n] + v_3[n] + v_4[n]$ (3)

$y[n] = (B_0 x[n] + (B_1 x[n-1] + (B_2 x[n-1]) + (B_3 x[n-3])$ (4)

By producing the approximate sum and breaking the CPC, the designed adder computes data in parallel. As a result, the FIR filter performs better than previous FIR filters are developed utilizing approximation adders and multipliers.

4. RESULT ANALYSIS

In this section, design and implementation of low power efficient FIR filter architectures using approximate adders and multipliers is presented. The table 1 shows performance evaluation.

Table 1. Performance Evaluation

Filter Type	Area (μm^2)	Delay (ns)	Power (mW)
FIR using Distributed Arithmetic (DA)	25658.98	1.02	12.69
Presented FIR filter using AA(Approximate Adder)	18233.3	0.234	5.3

Compared to FIR filter using DA, presented FIR filter has shown better performance. The figure 2 shows the delay performance comparison.

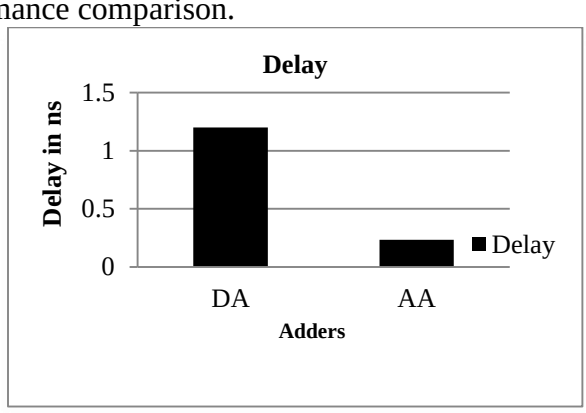


Figure 2. Delay Comparison

FIR filter using approximate multipliers and adders have very less delay than FIR using Distributed Arithmetic (DA). The Figure 3 shows power comparison.

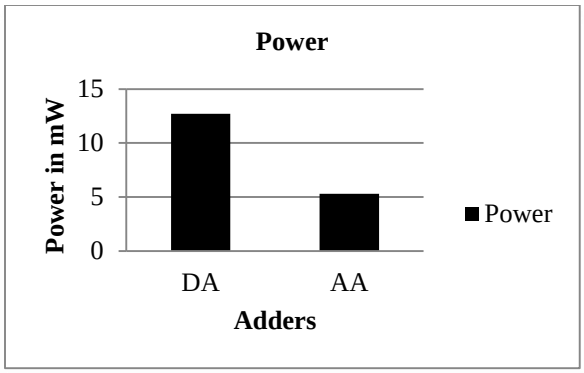


Figure 3. Power Comparison

Compared to DA, Arithmetic Adders (AA) has consumed very less power. The Figure 4 shows area comparison.

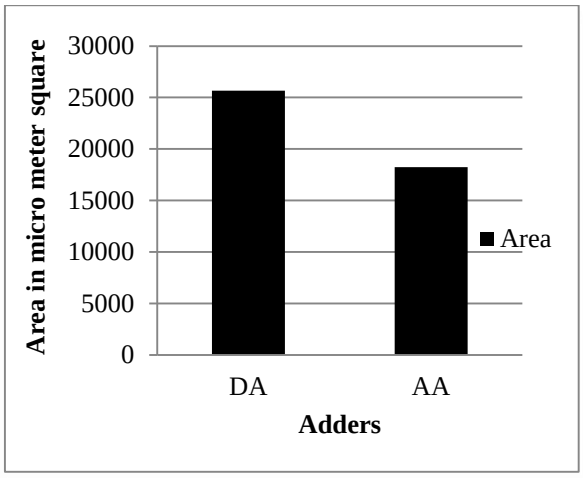


Figure 4. Area Comparison

Presented FIR filter using approximate multipliers and adder has very less area than FIR using Distributed Arithmetic. When compared to the conventional DA-based technique, the FIR filter implementation utilizing approximate multipliers and adders provided efficient values for the area, delay, and power.

5. CONCLUSION

In this work, the design and implementation of low power efficient FIR filter architectures using approximate adders and multipliers is presented. The analysis proposes approximate multiplier architectures using a simple and fast approximation adder. The created adder breaks the CPC is to calculate data in parallel while generating an approximate sum and error vector. To create FIR filters with excellent performance and low energy consumption, the approximation adder is suggested. The suggested fixes maintain higher performance while reducing delay and power in an efficient manner. By increasing speed and decreasing delay, approximate adder and multiplier sufficiently enhanced the FIR filter's performance. Comparing the suggested architecture to different architectures, they can conclude that it performs better in terms of area, power, and delay.

6. REFERENCES

[1] Thingujam Churchil Singh and Manoj Kumar, “Design and implementation of two-dimensional digital finite impulse response filter using very high speed integrated circuit hardware description language,” International Journal of Electrical and Computer

Engineering (IJECE), vol. 12, no. 4, pp. 3684-3691, August 2022, doi: 10.11591/ijece.v12i4.pp3684-3691

[2] Kiran Kumar Bhadavath and Z. Mary Livinsa, “Implementation of Efficient FIR Filter Architectures using Distributed Arithmetic Computation for Digital Channelizer of SDR,” 2022, doi: 10.21203/rs.3.rs-1724960/v1

[3] Charles Rajesh Kumar J, D. Vinod Kumar and M.A. Majid, “High-performance, energy-efficient, and memory-efficient FIR filter architecture utilizing 8x8 approximate multipliers for wireless sensor network in the Internet of Things,” Memories - Materials, Devices, Circuits and Systems, vol. 3, pp. 1-9, 2022, doi:10.1016/j.memori.2022.100017

[4] B. U. V. Prashanth, Mohammed Riyaz Ahmed and Manjunath R. Kounte, “Design and implementation of DA FIR filter for bio-inspired computing architecture,” International Journal of Electrical and Computer Engineering (IJECE), vol. 11, no. 2, pp. 1709-1718, April 2021, doi: 10.11591/ijece.v11i2.pp1709-1718

[5] Aaron Don M. Africa, John Arvin Mercado and Joshua Kenichi Sim, “Development of an adaptive finite impulse response filter optimization algorithm using rough set theory,” Indonesian Journal of Electrical Engineering and Computer Science, vol. 31, no. 1, pp. 238-247, July 2023, doi: 10.11591/ijeecs.v31.i1.pp238-247

[6] Venkata Krishna Odugu, Venkata Narasimhulu C and Satya Prasad K, “An efficient VLSI architecture of 2-D finite impulse response filter using enhanced approximate compressor circuits,” Int J Circ Theor Appl, vol. 49, no. 11, pp. 3653-3668, 2021, doi: 10.1002/cta.3114

[7] Gondhi Navabharat Reddy, Sruthi Setlem, V. Prakasam3 and D. Kiran Kumar, “Approximate arithmetic circuits,” International Journal of Reconfigurable and Embedded Systems (IJRES), vol. 9, no. 3, pp. 183-200, November 2020, doi: 10.11591/ijres.v9.i3.pp183-200

[8] P. J. Edavoor, S. Raveendran and A. D. Rahulkar, "Approximate Multiplier Design Using Novel Dual-Stage 4:2 Compressors," in *IEEE Access*, vol. 8, pp. 48337-48351, 2020, doi: 10.1109/ACCESS.2020.2978773.

[9] A. Towhidy, R. Omidi and K. Mohammadi, "On the Design of Iterative Approximate Floating-Point Multipliers," in *IEEE Transactions on Computers*, vol. 72, no. 6, pp. 1623-1635, 1 June 2023, doi: 10.1109/TC.2022.3216465.

[10] P V V Satyanarayana, M. Venkanna, S. Jayasri, J. Vasavi Kalyani and B. Ramjee, “Design and Implementation of Approximate 7:2 compressor based 16-bit Dadda multiplier using verilog,” International Journal of Recent Technology and Engineering (IJRTE), vol. 12, no. 1, pp. 39-48, May 2023, doi:10.35940/ijrte.A7570.0512123

[11] S. Nambi, U. A. Kumar, K. Radhakrishnan, M. Venkatesan and S. E. Ahmed, "DeBAM: Decoder-Based Approximate Multiplier for Low Power Applications," in *IEEE Embedded Systems Letters*, vol. 13, no. 4, pp. 174-177, Dec. 2021, doi: 10.1109/LES.2020.3045165.

[12] Usthulamuri Penchalaiah and V.G. Siva Kumar, “Low energy, long sustainable and high-speed FIR filter based on truncated multiplier with SCG-HSCG adder,” materials today, proceedings, vol. 61, no. 2, pp. 504-511, 2022, doi:10.1016/j.matpr.2021.12.507.

[13] M. H. Nassralla, N. Akl and Z. Dawy, "A Clustering-Based Approach for Designing Low Complexity FIR Filters," in *IEEE Signal Processing Letters*, vol. 28, pp. 299-303, 2021, doi: 10.1109/LSP.2021.3050916.

[14] A. Jiang, H. K. Kwan, Y. Tang and Y. Zhu, "Sparse FIR Filter Design via Partial 1-Norm Optimization," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 8, pp. 1482-1486, Aug. 2020, doi: 10.1109/TCSII.2019.2937343.

[15] S.Priyadharsini and S.Gnanamurugan, “Design of 4:2 Compressor for Parallel Distributed Arithmetic FIR Filter,” Indo-Iranian Journal of Scientific Research (IIJSR), vol. 2, no. 4, pp. 50-57, Oct-Dec 2018, doi:10.2139/issrn.3531562

[16] Usha Jadhav and K.Priyanka Sahu, “Design and Implementation of FIR Filter Structure using High Adders and Wallace Tree Multiplier,” International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 6, no. 6, pp. 4590-4596, June 2017, doi:10.15662/IJAREEIE.2017.0606068,

[17] Anusha Gorantla and Deepa P, “Design of approximate compressors for multiplication,” J. Emerg. Technol. Comput. Syst, vol.13, no. 3, pp. 1-17, April 2017, doi: 10.1145/3007649.

[18] T. Thanmai and J. Ravindra, "High Performance, Low Power Architecture of 5-stage FIR Filter using Modified Montgomery Multiplier," 2020 International Conference on Applied Electronics (AE), Pilsen, Czech Republic, 2020, pp. 1-5, doi: 10.23919/AE49394.2020.9232853.

[19] Wenbin Ye, “Multiplierless Multiple-Stage Cascaded FIR Filter Design,” Journal of Circuits, Systems, and Computers, vol. 24, no. 1, pp. 1-7, 2015, doi: 10.1142/S0218126615500115

[20] S. Madhavi, K. Rasagna, N. Kavya, M. Sindhu and V. Kaveri, "Implementation of Programmable FIR Filter Using Dadda Multiplier and Parallel Prefix Adder," 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2018, pp. 585-589, doi: 10.1109/ICIRCA.2018.8597249.

[21] W. Chen, M. Huang and X. Lou, "Design of Sparse FIR Filters With Reduced Effective Length," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 66, no. 4, pp. 1496-1506, April 2019, doi: 10.1109/TCSI.2018.2883965.

[22] J. Chen, C. H. Chang, J. Ding, R. Qiao and M. Faust, "Tap Delay-and-Accumulate Cost Aware Coefficient Synthesis Algorithm for the Design of Area-Power Efficient FIR Filters," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 65, no. 2, pp. 712-722, Feb. 2018, doi: 10.1109/TCSI.2017.2725916.

[23] O. Venkata Krishna, “Design and Implementation of FIR Filter using Low Power and High Speed Multiplier and Adders**,”** CVR Journal of Science and Technology, vol. 17, pp. 127-131, December 2019, doi: 10.32377/cvrjst1722

[24] Sunesh N.V and Sathishkumar P, "Design and implementation of fast floating point multiplier unit," *2015 International Conference on VLSI Systems, Architecture, Technology and Applications (VLSI-SATA)*, Bengaluru, India, 2015, pp. 1-5, doi: 10.1109/VLSI-SATA.2015.7050478.

[25] S. Jayakumar, R.Selvam, K. Karthikeyan and R. Natesan, “Design and Implementation of Hybrid FIR Filters using Vedic Multiplier and Fast Adders,” International Journal of Recent Technology and Engineering (IJRTE), vol.8, no. 4, pp. 11849-11853, November 2019, doi: 10.35940/ijrte.D9569.118419

[26] G. Capizzi and G. L. Sciuto, "A Novel 2-D FIR Filter Design Methodology Based on a Gaussian-Based Approximation," in *IEEE Signal Processing Letters*, vol. 26, no. 2, pp. 362-366, Feb. 2019, doi: 10.1109/LSP.2019.2891965.