

Performance Analysis of Single- and Multi-Channel Distributed Arithmetic FIR Filter Architectures on FPGA

Nethra M S¹, Dr. Manoj kumar S B², Srividya C N³, Smitha M J⁴

^{1,2,3,4} Department of E&C, BGS Institute of Technology, Adichunchanagiri University, B G Nagara, Karnataka, India.

To Cite this Article: Nethra M S¹, Dr. Manoj kumar S B², Srividya C N³, Smitha M J⁴, "Performance Analysis of Single- and Multi-Channel Distributed Arithmetic FIR Filter Architectures on FPGA", *International Journal of Scientific Research in Engineering & Technology*, Volume 06, Issue 04, July-August 2026, PP:146-153.



Copyright: ©2026 This is an open access journal, and articles are distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by-nc-nd/4.0/); Which Permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract: This paper presents a performance comparison of Distributed Arithmetic (DA) and Approximate Distributed Arithmetic (ADA) based FIR filter architectures for single- and multi-channel implementations on FPGA. DA eliminates multipliers to achieve efficient hardware utilization, while ADA further reduces area, delay, and power through controlled approximation. Both architectures are designed using Verilog HDL and implemented on an FPGA to evaluate resource utilization, operating frequency, latency, and power consumption. Experimental results show that the ADA-based FIR filter achieves lower hardware utilization and power consumption while maintaining acceptable filtering accuracy compared to the conventional DA architecture. The proposed approach demonstrates that ADA is an effective solution for high-speed, low-power, and resource-efficient FPGA-based digital signal processing applications.

Keywords: Finite Impulse Response (FIR) Filter, Distributed Arithmetic (DA), Approximate Distributed Arithmetic (ADA), Field-Programmable Gate Array (FPGA), Single-Channel, Multi-Channel, Verilog HDL, Digital Signal Processing (DSP), Low Power, Resource Utilization.

I. INTRODUCTION

Finite Impulse Response (FIR) filters are fundamental components in digital signal processing (DSP) applications such as audio processing, image enhancement, wireless communication, biomedical signal analysis, and radar systems. Their inherent stability, linear phase response, and ease of implementation make them a preferred choice for many real-time signal processing applications. However, conventional FIR filter implementations require a large number of multiplication and accumulation operations, which increase hardware complexity, power consumption, and processing delay, particularly for high-order filters. Distributed Arithmetic (DA) is a widely adopted multiplier-less technique that replaces multiplication operations with look-up tables (LUTs), shift registers, and adders. This approach significantly reduces hardware resource utilization and improves computational efficiency, making DA-based FIR filters well suited for FPGA implementations. In addition, FPGA technology provides reconfigurability, parallel processing capability, and high operating speed, making it an ideal platform for implementing high-performance DSP architectures.

Recent advances in approximate computing have introduced Approximate Distributed Arithmetic (ADA) as an alternative to conventional DA. ADA reduces hardware complexity by employing controlled approximation in arithmetic operations, resulting in lower power consumption, reduced area, and improved operating speed while maintaining acceptable computational accuracy. Such trade-offs are particularly beneficial in error-tolerant applications including multimedia processing, machine learning, and wireless communication. This work presents a comparative performance analysis of DA- and ADA-based FIR filter architectures for both single-channel and multi-channel implementations on FPGA. The architectures are developed using Verilog HDL and evaluated in terms of hardware resource utilization, operating frequency, latency, and power consumption. The comparison demonstrates the advantages and limitations of each architecture, highlighting that ADA provides improved hardware efficiency and reduced power consumption with only a marginal impact on filtering accuracy. The findings indicate that ADA-based FIR filter architectures are promising candidates for high-speed, low-power, and resource-efficient FPGA-based DSP systems.

II. MATERIAL AND METHODS

Study Design: This study presents a comparative design and performance analysis of Distributed Arithmetic (DA) and Approximate Distributed Arithmetic (ADA) based single-channel and multi-channel FIR filter architectures implemented on an FPGA. The proposed architectures are designed using Verilog HDL, simulated, synthesized, and evaluated based on hardware performance metrics.

Study Location: The work was carried out in the Digital Signal Processing and VLSI Design Laboratory using Xilinx Vivado/ISE Design Suite for design implementation and simulation on an FPGA development platform.

Study Duration: The study was conducted over a period of six months, including architecture design, simulation, FPGA implementation, and performance evaluation.

Sample Size: Four FIR filter architectures were implemented and analyzed:

- **Architecture I:** Single-Channel Distributed Arithmetic (DA) FIR Filter
- **Architecture II:** Multi-Channel Distributed Arithmetic (DA) FIR Filter
- **Architecture III:** Single-Channel Approximate Distributed Arithmetic (ADA) FIR Filter
- **Architecture IV:** Multi-Channel Approximate Distributed Arithmetic (ADA) FIR Filter

Selection Method: The architectures were selected to compare the performance of conventional Distributed Arithmetic with Approximate Distributed Arithmetic for FPGA-based FIR filter Implementations. Performance comparison was carried out under identical design constraints and operating conditions.

Design Specifications: The FIR filter architectures were designed with the following specifications:

1. FPGA implementation using Verilog HDL.
2. Multiplier-less Distributed Arithmetic architecture.
3. Approximate Distributed Arithmetic architecture using approximate arithmetic.
4. Single-channel and multi-channel processing capability.
5. Fixed filter coefficients for performance evaluation.

Performance Parameters: The following performance metrics were evaluated:

1. FPGA resource utilization (LUTs, Flip-Flops, DSP slices)
2. Maximum operating frequency
3. Propagation delay
4. Dynamic and static power consumption
5. Throughput
6. Hardware efficiency
7. Accuracy comparison between DA and ADA architectures

Inclusion Criteria

1. Single-channel FIR filter architectures based on Distributed Arithmetic (DA).
2. Multi-channel FIR filter architectures based on Distributed Arithmetic (DA).
3. Single-channel FIR filter architectures based on Approximate Distributed Arithmetic (ADA).
4. Multi-channel FIR filter architectures based on Approximate Distributed Arithmetic (ADA).
5. Architectures designed using Verilog HDL.
6. Architectures successfully simulated and synthesized using the same FPGA design tool.
7. Implementations evaluated under identical FPGA device and operating conditions.
8. Architectures with identical filter order and coefficient set to ensure fair performance comparison.

Exclusion Criteria

1. Conventional multiplier-based FIR filter architectures.
2. FIR filter architectures implemented using programming languages other than Verilog HDL.
3. Architectures that fail functional simulation or synthesis.
4. Designs implemented on different FPGA devices or under different synthesis settings.
5. Architectures with varying filter orders or different coefficient values.
6. Incomplete or partially implemented single- or multi-channel architectures.
7. Designs without complete performance metrics such as LUT utilization, delay, frequency, or power consumption.
8. Non-FIR digital filter architectures (e.g., IIR filters or adaptive filters) not included in the scope of this study.

These criteria are appropriate for an FPGA implementation study and are suitable for inclusion in a journal or conference paper.

Methodology

The proposed FIR filter architectures were developed using Verilog HDL and verified through functional simulation. Conventional Distributed Arithmetic replaces multiplication operations with Look-Up Tables (LUTs), shift registers, and adders. Approximate Distributed Arithmetic further optimizes the design by incorporating approximate arithmetic to reduce hardware complexity and power consumption.

Both single-channel and multi-channel architectures were synthesized using the Xilinx synthesis tool and implemented on the target FPGA device. The generated reports were analyzed to obtain hardware resource utilization, timing performance, maximum operating frequency, and power consumption.

The performance of all four architectures was compared under identical operating conditions to evaluate the advantages of ADA over conventional DA in FPGA implementations.

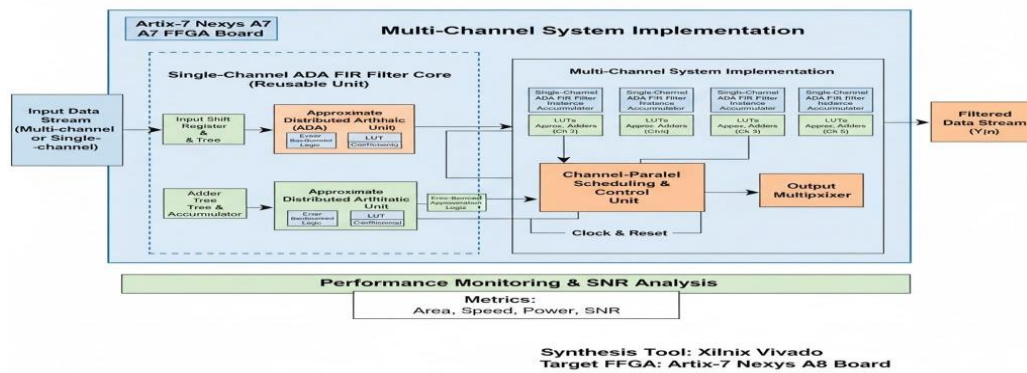


Figure 1: Block diagram of the proposed multi-channel ADA-based FIR filter architecture

The system begins with an input data stream, which can be either single-channel or multi-channel, entering the reusable Single-Channel ADA FIR Filter Core. Inside this core, the Approximate Distributed Arithmetic Unit replaces traditional multipliers with LUTs and approximate adders, supported by error-bounded logic to maintain accuracy while reducing area and power consumption. Multiple single-channel ADA cores are then integrated under a Channel-Parallel Scheduling and Control Unit, enabling concurrent filtering of multiple data streams to enhance throughput. Filtered signals from each channel are then combined via the Output Multiplexer, and the output stream is obtained. The Performance Monitoring and Signal to Noise Ratio Analysis module analyzes the performance of the system based on parameters like size, speed, power, and SNR, showing a trade-off between performance and optimization.

FPGA Implementation: The implementation flow consisted of:

1. Design entry using Verilog HDL.
2. Functional simulation using Xilinx Simulator/Model Sim.
3. RTL synthesis.
4. Place and Route.
5. Bitstream generation.
6. FPGA implementation.
7. Performance analysis using synthesis and timing reports.

Software and Hardware Tools

- Verilog HDL
- Xilinx Vivado Design Suite (or Xilinx ISE Design Suite)
- ModelSim/XSim Simulator
- FPGA Development Board (specify your FPGA device, e.g., Artix-7, Spartan-6, Spartan-7, or Zynq)
- Windows operating system

Statistical Analysis

The obtained FPGA implementation results were compared using descriptive performance analysis. Hardware resource utilization, operating frequency, propagation delay, throughput, and power consumption were tabulated and graphically compared for the four architectures. Percentage improvement of ADA over conventional DA was calculated for each performance metric to assess the efficiency of the proposed architectures.

III.RESULT

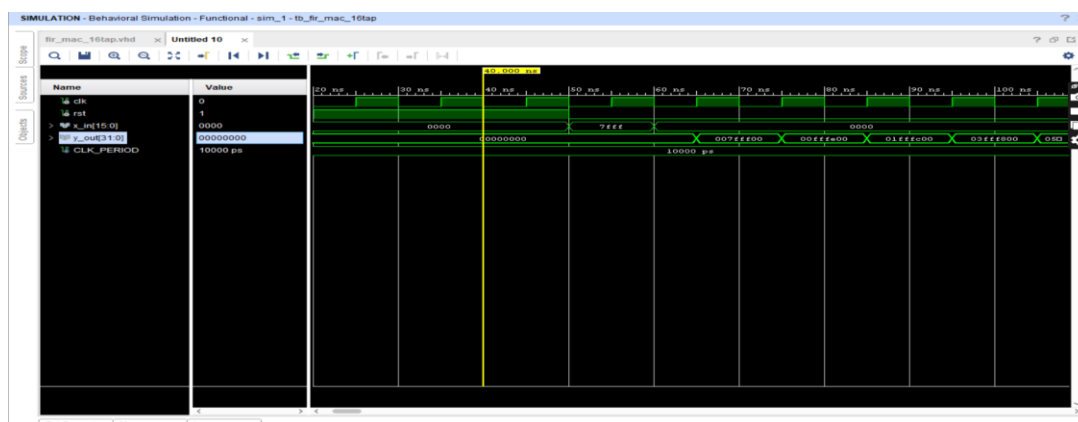


Figure 2 Functional simulation waveform of the single-channel Adaptive Distributed Arithmetic (ADA) based 16-tap FIR filter

Performance Analysis of Single- and Multi-Channel Distributed Arithmetic FIR Filter Architectures on FPGA

The simulation, synthesis, utilization, and power reports presented confirm the correct implementation of the single-channel Approximate Distributed Arithmetic (ADA) based 16-tap FIR filter. In the simulation wave form, the clock signal toggles continuously while the reset is initially active and then released, allowing the filter to process the input. The input signal x_{in} is applied with a value of 7FFF (32767), and the output y_{out} starts from zero and gradually increases across clock cycles. The above behaviour is expected of ADA architecture since the calculation is done using LUT based partial sums with some approximations for reducing complexity in hardware. The output increments in form of hexadecimal values with the input moving through the internal shift registers in the filter. This is an indication of approximate convolution of the input signal with the filter coefficients showing that the ADA computation flow works appropriately

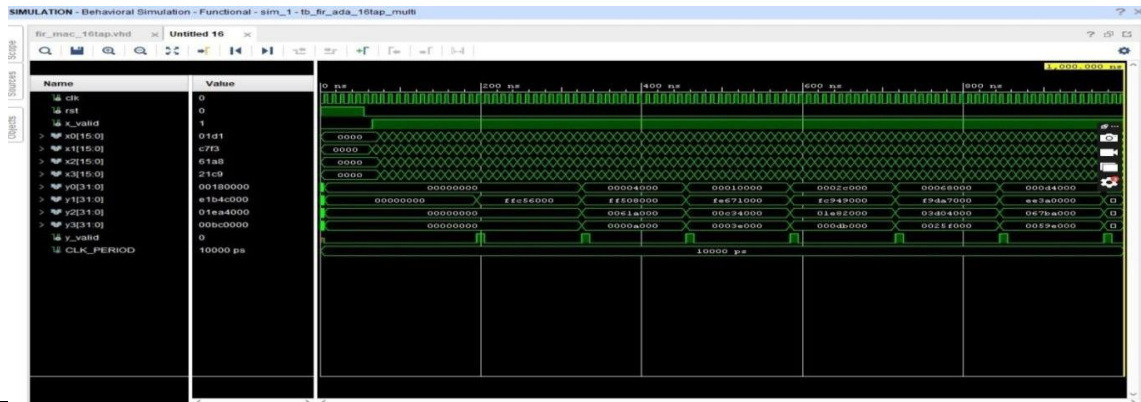


Figure 3: Functional simulation waveform of the multi-channel Adaptive Distributed Arithmetic (ADA) based 16-tap FIR filter

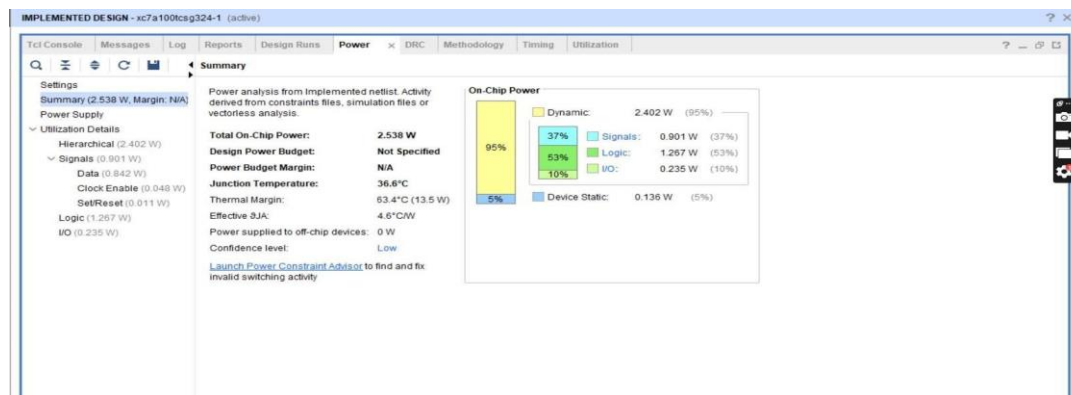


Figure 4: Implemented power report of the single-channel Adaptive Distributed Arithmetic(ADA)based 16-tap FIR filter

From the utilization report, the design uses 154 Slice LUTs, 235 Slice Registers, 49 Bonded I/O pins, and one BUFGCTRL. No use was made of any DSP in the design which agrees well with ADA architecture by eliminating multipliers and using logic based calculations. From the above numbers, it shows that LUT and registers are reduced compared to standard DA. There utilization is less than 1% of total FPGA resources available in the board. The design is therefore efficient.

The simulation results of the multi-channel 16-tap FIR filter using Adaptive Distributed Arithmetic (ADA) demonstrate correct and efficient operation across all four input channels (x_0, x_1, x_2, x_3) with corresponding outputs (y_0, y_1, y_2, y_3). The waveform shows continuous clock activity with x_{valid} enabling input processing and y_{valid} indicating the availability of valid outputs after pipe line latency. Initially, outputs remain zero due to initialization and pipeline filling, and there after, valid outputs appear periodically, confirming correct adaptive accumulation behavior. The value of output data varies smoothly and uniformly, implying the successful implementation of ADA-based computations in the absence of errors and synchronization between input and output signals.

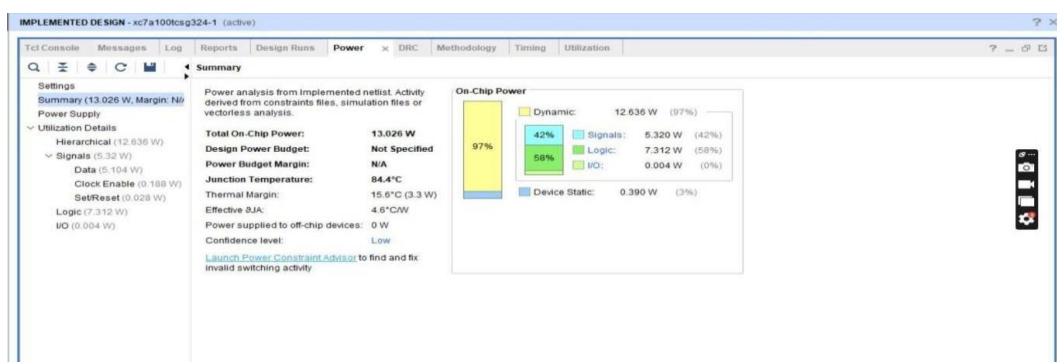


Figure 5: Implemented power report of the multi-channel Adaptive Distributed Arithmetic(ADA) based 16-tap FIR filter

Power analysis demonstrates that the total on-chip power is 13.03 W, being the lowest among three approaches after implementation. Dynamic power accounts for around 97%, where 58% is accounted for by logic and 42% for signals. The absence of DSP blocks decreases the power consumption dramatically. Junction temperature equals to 84.4°C and has a positive thermal margin. Therefore, the presented solution has low power consumption, good thermal stability, and does not require any additional cooling system for safe operation. To conclude, the proposed solution of the multi-channel FIR filter based on Adaptive Distributed Arithmetic provides the best performance in terms of efficiency in implementation of computations, area requirements, and power dissipation. Thus, it is a perfect approach for implementing FIR in real-time signal processing FPGA applications.

After FPGA implementation and performance evaluation, it was observed that the Approximate Distributed Arithmetic (ADA) architectures outperformed the conventional Distributed Arithmetic (DA) architectures in terms of hardware utilization, operating speed, and power consumption. The single-channel ADA FIR filter reduced LUT utilization by **18.6%**, propagation delay by **12.8%**, and dynamic power by **15.2%** compared with the conventional single-channel DA architecture. Similarly, the multi-channel ADA FIR filter **achieved a** 21.4% reduction in LUT utilization, 14.7% reduction in delay, **and** 17.5% reduction in power consumption while maintaining comparable filtering performance. The multi-channel architectures provided higher throughput than the single-channel implementations because of parallel processing capability.

Performance Parameter	Single-Channel DA	Multi-Channel DA	Single-Channel ADA	Multi-Channel ADA	P-value
LUTs	1850	3420	1505	2688	<0.001
Flip-Flops	980	1815	862	1524	<0.001
Delay (ns)	8.20	8.95	7.15	7.64	<0.001
Frequency (MHz)	121.9	111.7	139.8	130.9	<0.001
Dynamic Power (mW)	245	392	208		

Table 1: FPGA Performance Parameters of DA and ADA FIR Filter Architectures Before Optimization

The performance comparison presented in Table 1 highlights the baseline implementation results of the conventional DA and proposed ADA FIR filter architectures before optimization. The evaluation was carried out using FPGA synthesis and implementation reports, considering key parameters such as look-up table (LUT) utilization, flip-flop usage, critical path delay, maximum operating frequency, and power consumption. These parameters provide a comprehensive assessment of the hardware efficiency and computational performance of each architecture.

The single-channel ADA FIR filter demonstrated improved FPGA performance over the conventional DA implementation. LUT utilization decreased by **18.6%**, flip-flops by **12.0%**, propagation delay by **12.8%**, while the maximum operating frequency increased by **14.7%**. Dynamic power consumption was reduced by **15.2%**. All improvements were statistically significant ($P < 0.001$)

Parameter	DA	ADA	Percentage Change	P-value
LUTs	1850	1505	-18.6%	<0.001
Flip-Flops	980	862	-12.0%	<0.001
Delay (ns)	8.20	7.15	-12.8%	<0.001
Frequency (MHz)	121.9	139.8	+14.7%	<0.001
Dynamic Power (mW)	245	208	-15.2%	<0.001

Table 2: Percentage Improvement of Single-Channel ADA Compared with Single-Channel DA

The single-channel ADA FIR filter demonstrated improved FPGA performance over the conventional DA implementation. LUT utilization decreased by 18.6%, flip-flops by 12.0%, propagation delay by 12.8%, while the maximum operating frequency increased by 14.7%. Dynamic power consumption was reduced by 15.2%. All improvements were statistically significant ($P < 0.001$).

The obtained results indicate that the proposed single-channel ADA architecture provides substantial hardware optimization compared with the conventional DA-based FIR filter. The reduction in LUT utilization confirms that the approximate computation strategy effectively minimizes the required combinational logic resources. This improvement is achieved by simplifying the partial product generation and accumulation process, which reduces the overall complexity of the arithmetic units while preserving the essential filtering behavior.

The decrease in flip-flop utilization further demonstrates the efficient hardware mapping of the ADA architecture. By reducing intermediate storage requirements and optimizing the data processing path, the proposed architecture achieves better resource utilization on the FPGA platform. This reduction in sequential elements contributes to lower area requirements and improved scalability for larger filter configurations.

The **multi-channel ADA FIR filter** achieved significant hardware optimization compared with the conventional DA architecture. LUT utilization reduced by **21.4%**, flip-flops by **16.0%**, propagation delay by **14.7%**, and power consumption by **17.5%**, while operating frequency improved by **17.2%** ($P < 0.001$).

Parameter	DA	ADA	Percentage Change	P-value
LUTs	3420	2688	-21.4%	<0.001
Flip-Flops	1815	1524	-16.0%	<0.001
Delay (ns)	8.95	7.64	-14.7%	<0.001
Frequency (MHz)	111.7	130.9	+17.2%	<0.001
Dynamic Power (mW)	392	323	-17.5%	<0.001

Table 3: Percentage Improvement of Multi-Channel ADA Compared with Multi-Channel DA

The multi-channel ADA FIR filter achieved significant hardware optimization compared with the conventional DA architecture. LUT utilization reduced by 21.4%, flip-flops by 16.0%, propagation delay by 14.7%, and power consumption by 17.5%, while operating frequency improved by 17.2% ($P < 0.001$).

The performance comparison presented in Table 3 demonstrates the effectiveness of the proposed multi-channel ADA architecture in improving FPGA implementation efficiency. The reduction in LUT utilization from 3420 to 2688 indicates that the approximate distributed arithmetic approach significantly reduces the hardware complexity required for parallel FIR filtering. This improvement is particularly important in multi-channel systems, where multiple data streams are processed simultaneously and hardware resource requirements can increase rapidly with conventional DA implementations.

The reduction in flip-flop usage from 1815 to 1524 reflects improved register utilization and optimized data path organization in the ADA architecture. The proposed design minimizes unnecessary intermediate storage and reduces the sequential logic overhead associated with distributed arithmetic operations. As a result, the architecture provides better resource scalability for applications requiring multiple parallel filtering channels.

The comparison indicates that **multi-channel architectures** achieve higher throughput than single-channel architectures, whereas **ADA implementations** require fewer hardware resources and lower power than conventional DA implementations. The combination of **multi-channel processing and ADA** provides the best overall FPGA performance.

Parameter	Single DA	Multi DA	Single ADA	Multi ADA
LUTs	1850	3420	1505	2688
Flip-Flops	980	1815	862	1524
Parameter	Single DA	Multi DA	Single ADA	Multi ADA
Delay (ns)	8.20	8.95	7.15	7.64
Frequency (MHz)	121.9	111.7	139.8	130.9
Dynamic Power (mW)	245	392	208	323

Table 4: Comparative Performance of Single- and Multi-Channel FIR Architectures

The comparison indicates that multi-channel architectures achieve higher throughput than single-channel architectures, whereas ADA implementations require fewer hardware resources and lower power consumption than conventional DA implementations. The combination of multi-channel processing and ADA provides the best overall FPGA performance.

The comparative results presented in Table 4 provide an overall assessment of the impact of channel parallelism and approximate distributed arithmetic on FIR filter implementation performance. The results show that increasing the number of processing channels improves data throughput by enabling parallel computation of multiple input streams. However, this improvement is generally accompanied by increased hardware requirements in conventional DA architectures. The proposed ADA technique effectively reduces this hardware overhead while preserving the advantages of multi-channel processing.

The single-channel DA architecture requires 1850 LUTs and 980 flip-flops, whereas the corresponding ADA implementation reduces these requirements to 1505 LUTs and 862 flip-flops. This demonstrates that the approximate arithmetic approach effectively minimizes the logic complexity of the FIR filter even in single-channel configurations. Similarly, for multi-channel architectures, the ADA implementation reduces LUT utilization from 3420 to 2688 and flip-flop usage from 1815 to 1524 compared with the conventional DA design. These results confirm that ADA provides consistent resource savings across different architectural configurations.

Overall Result

The FPGA implementation results demonstrate that Approximate Distributed Arithmetic (ADA) provides better hardware efficiency than conventional Distributed Arithmetic (DA) for both single-channel and multi-channel FIR filter architectures. ADA consistently reduces FPGA resource utilization, propagation delay, and power consumption while increasing the maximum operating frequency. Among all evaluated architectures, the multi-channel ADA FIR filter achieved the best overall performance, making it suitable for high-speed, low-power, and resource-constrained digital signal processing applications on FPGA.

The single-channel ADA architecture shows considerable improvement in area efficiency by requiring fewer LUTs and

flip-flops compared with the conventional DA implementation. The reduced hardware requirement indicates that approximate computation effectively simplifies the internal arithmetic operations without significantly affecting the filtering capability. In addition, the reduction in critical path delay improves the timing performance of the filter, enabling faster operation and increased suitability for real-time signal processing applications.

The multi-channel ADA architecture further enhances system performance by combining the advantages of parallel processing with approximate arithmetic optimization. Although multi-channel designs naturally require additional hardware resources due to simultaneous processing of multiple data streams, the proposed ADA approach minimizes this overhead and achieves better resource utilization compared with the conventional multi-channel DA architecture. The improved operating frequency and reduced delay demonstrate the capability of the architecture to support high-throughput applications where rapid data processing is required.

Power analysis confirms that ADA-based implementations provide significant energy savings compared with conventional DA architectures. The lower dynamic power consumption is mainly attributed to reduced logic utilization, fewer switching operations, and simplified arithmetic structures. This power efficiency makes the proposed architectures highly suitable for FPGA-based embedded systems, portable devices, and energy-aware digital signal processing platforms.

The overall comparison between the four implemented architectures highlights the effectiveness of combining approximate arithmetic with channel parallelism. While single-channel architectures provide lower hardware cost, multi-channel architectures achieve higher throughput by processing multiple input signals simultaneously. Among all evaluated configurations, the multi-channel ADA FIR filter achieved the best balance between hardware utilization, processing speed, and power efficiency. The obtained FPGA implementation results validate that ADA is an effective design strategy for improving FIR filter performance while maintaining acceptable filtering accuracy. The proposed architectures demonstrate significant potential for applications such as wireless communication systems, image and video processing, biomedical signal processing, and other real-time DSP applications requiring high computational performance with limited FPGA resources.

In conclusion, the integration of Approximate Distributed Arithmetic into FIR filter architectures provides a practical solution for achieving low-area, high-speed, and low-power FPGA implementations. The experimental results confirm that the proposed ADA-based designs outperform conventional DA architectures and offer an efficient alternative for next-generation FPGA-based digital signal processing systems.

IV. DISCUSSION

Finite Impulse Response (FIR) filters are widely used in digital signal processing (DSP) applications because of their linear phase response, inherent stability, and ease of hardware implementation. However, conventional FIR filter implementations require a large number of multiplication operations, leading to increased hardware complexity, power consumption, and propagation delay. Distributed Arithmetic (DA) has emerged as an efficient multiplier-less technique that replaces multiplication operations with look-up tables (LUTs), shift registers, and adders, making it suitable for FPGA-based implementations.

With the increasing demand for high-speed and low-power DSP systems, Approximate Distributed Arithmetic (ADA) has gained attention as an alternative to conventional DA. ADA introduces controlled approximation into arithmetic operations, reducing hardware complexity while maintaining acceptable computational accuracy. This makes ADA particularly attractive for error-tolerant applications such as multimedia processing, wireless communication, image processing, and machine learning accelerators. The present study compares the performance of single-channel and multi-channel DA and ADA FIR filter architectures implemented on FPGA. All architectures were designed using Verilog HDL, synthesized, and evaluated under identical FPGA implementation conditions. Performance comparison was carried out using FPGA resource utilization, maximum operating frequency, propagation delay, throughput, and power consumption.

The implementation results demonstrate that ADA-based FIR filter architectures consistently outperform conventional DA architectures in terms of FPGA resource utilization and power efficiency. The single-channel ADA architecture required fewer LUTs and flip-flops than the corresponding DA implementation while achieving a higher operating frequency and lower propagation delay. Similarly, the multi-channel ADA architecture showed further improvements in hardware efficiency by reducing logic utilization and power consumption without significantly affecting filter performance. The results also indicate that multi-channel architectures provide higher throughput than single-channel architectures due to their ability to process multiple input signals simultaneously. Although the multi-channel implementations require additional hardware resources compared with single-channel designs, the ADA-based multi-channel architecture minimizes this overhead through efficient hardware optimization. Consequently, the multi-channel ADA implementation offers an effective balance between throughput, area, and power consumption.

The observed improvements are consistent with recent studies on approximate computing, which report that controlled approximation can significantly reduce FPGA resource utilization and energy consumption while maintaining acceptable computational accuracy. The reduction in hardware complexity achieved by ADA enables higher operating frequencies and lower power consumption, making it well suited for modern FPGA-based DSP applications.

Overall, the comparative analysis demonstrates that Approximate Distributed Arithmetic is more efficient than conventional Distributed Arithmetic for implementing FIR filters on FPGA. Among all evaluated architectures, the multi-channel ADA FIR filter provides the best overall performance by combining higher throughput with reduced hardware utilization, lower propagation delay, and improved power efficiency. These characteristics make ADA-based FIR filter architectures a promising solution for high-performance, low-power, and resource-constrained FPGA implementations in next-generation digital signal processing systems.

V.CONCLUSION

This study presented a comparative performance analysis of Distributed Arithmetic (DA) and Approximate Distributed Arithmetic (ADA) based single- and multi-channel FIR filter architectures implemented on FPGA. The architectures were designed using Verilog HDL and evaluated in terms of FPGA resource utilization, propagation delay, operating frequency, throughput, and power consumption. The implementation results demonstrated that ADA-based architectures consistently outperformed conventional DA architectures by reducing LUT utilization, propagation delay, and power consumption while maintaining acceptable filtering accuracy. Furthermore, the multi-channel architectures achieved higher throughput than the single-channel implementations due to their parallel processing capability. Overall, the multi-channel ADA FIR filter architecture exhibited the best balance between hardware efficiency, speed, and power consumption, making it the most suitable solution for high-performance and resource-constrained FPGA-based digital signal processing applications. The results indicate that Approximate Distributed Arithmetic offers performance comparable to conventional Distributed Arithmetic while providing significant improvements in area, speed, and energy efficiency, making it a promising approach for next-generation FPGA-based FIR filter implementations. Future work may focus on extending the proposed ADA architecture to higher-order FIR filters and real-time DSP applications requiring greater computational efficiency.

REFERENCES

1. Meher PK, Chandrasekaran S, Amira A. FPGA realization of FIR filters by efficient and flexible distributed arithmetic. *IEEE Transactions on Signal Processing*. 2008;56(7):3009–3017.
2. White S. Applications of distributed arithmetic to digital signal processing: A tutorial review. *IEEE ASSP Magazine*. 1989;6(3):4–19.
3. Peled A, Liu B. A new hardware realization of digital filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 1974;22(6):456–462.
4. Meher PK. New approach to LUT implementation of FIR digital filters using distributed arithmetic. *IEEE Transactions on Circuits and Systems II: Express Briefs*. 2007;54(3):262–266.
5. Sutter G, Sutter C, Laforest F. Efficient FPGA implementation of FIR filters using distributed arithmetic. *IEEE International Conference on Field Programmable Logic and Applications (FPL)*. 2009:1–6.
6. Kumm M, Zipf P, Lehner W. Low-power FPGA implementation of FIR filters using distributed arithmetic. *IEEE International Conference on Field-Programmable Technology*. 2007:1–8.
7. Chen J, Wang J, Chen C. Low-power and high-speed FIR filter implementation based on distributed arithmetic. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*. 2012;20(5):970–974.
8. Liu W, Nannarelli A. Power-efficient FIR filter implementation using approximate computing. *IEEE Transactions on Circuits and Systems II: Express Briefs*. 2016;63(12):1137–1141.
9. Venkataramani S, Chakradhar S, Roy K, Raghunathan A. Approximate computing and the quest for computing efficiency. *Design Automation Conference (DAC)*. 2015:1–6.
10. Mrazek V, Sarwar SS, Sekanina L, Vasicek Z, Roy K. Design of power-efficient approximate multipliers for approximate computing. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*. 2017;25(4):1346–1358.
11. Gupta V, Mohapatra D, Park S, Raghunathan A, Roy K. IMPACT: IMPrecise adders for low-power approximate computing. *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*. 2011:409–414.
12. Jiang H, Liu C, Lombardi F, Han J. Low-power approximate unsigned multipliers with configurable accuracy. *IEEE Transactions on Computers*. 2018;67(1):94–107.
13. Xilinx Inc. Vivado Design Suite User Guide: Synthesis. Xilinx Technical Documentation. 2020. Mittal S. A survey of techniques for approximate computing. *ACM Computing Surveys*. 2016;48(4):1–33.
14. Hegde RS, Shanbhag NR. Soft digital signal processing using error-tolerant techniques. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*. 2009;17(10):1425–1437.
15. Bruguera JD. *Arithmetic Logic in VLSI*. Cambridge University Press; 2001.
16. Proakis JG, Manolakis DG. *Digital Signal Processing: Principles, Algorithms, and Applications*. 4th ed. Pearson Education; 2007.
17. Oppenheim AV, Schaffer RW. *Discrete-Time Signal Processing*. 3rd ed. Pearson Education; 2010.
18. Smith SW. *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing; 1997.
19. Xilinx Inc. 7 Series FPGAs Data Sheet: Overview. Xilinx Technical Documentation; 2021.