International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

# Implementation of FIR Filters Through Inner Product Units and Parallel Accumulations

**Adabala Lahari 1, Josh Joseph Empra 2, Ranga Sai Teja 3, Ms. Heena Kauser 4**

[1,2,3] UG Scholar, Dept. Of ECE, St. Martin's Engineering College, Secunderabad, Telangana, India, 500100
[4]Assistant Professor, Dept. Of ECE, St. Martin's Engineering College, Secunderabad, Telangana, India, 50010
lahariadabala97@gmail.com

***Abstract:***

Finite Impulse Response (FIR) filters are essential components in digital signal processing (DSP) applications due to their stability and linear phase characteristics. However, traditional FIR filter designs often face challenges such as high-power consumption, increased area utilization, and significant delays, particularly in high-performance applications. This paper proposes an optimized FIR filter architecture utilizing inner product units and parallel accumulation techniques to address these challenges. The proposed design incorporates pipelined adders, coefficient storage units, and parallel processing to achieve significant improvements in power efficiency, area optimization, and computational speed. Experimental results obtained using Xilinx Vivado demonstrate a 30% reduction in LUT usage, a 20% decrease in power consumption, and a 27.2% improvement in delay metrics compared to existing systems. The enhanced architecture is ideal for real-time DSP applications in fields such as communications, biomedical signal processing, and audio systems.

**Keywords: FIR Filter, Digital Signal Processing (DSP), Inner Product Units (IPUs), Parallel Accumulation, Pipelined Architecture, Power Optimization, Area Efficiency, Setup Delay Reduction, LUT Utilization, VLSI Implementation, High-Speed Computing, Real-Time Processing, Biomedical Signal Processing,Communication Systems, Noise Reduction, Distributed Arithmetic.**

## 1. INTRODUCTION

FIR filters are a cornerstone of digital signal processing (DSP) and are widely used in numerous applications, including radar, sonar, seismology, wireless communications, radio astronomy, acoustics, and biomedicine. These filters are particularly valued for their simplicity and flexibility, as they offer the ability to design filters with precise frequency responses while maintaining excellent stability and linear phase characteristics. An FIR filter is characterized by a finite-length impulse response, meaning it responds to an input signal for only a limited duration before settling to zero. This contrasts with infinite impulse response (IIR) filters, which have internal feedback and an impulse response that continues indefinitely. The finite nature of an FIR filter's response makes it inherently stable and prevents issues of instability that can arise with IIR filters.

The implementation of FIR filters typically involves using a series of delays, multipliers, and adders to compute the filter's output. This straightforward structure makes FIR filters easier to implement in hardware and software, and they are less prone to errors that can accumulate over time, as no feedback is involved. This also means that FIR filters do not suffer from the potential for instability that can occur in IIR filters due to feedback loops. Furthermore, because FIR filters are non-recursive, the coefficients used in their design can be adjusted to achieve specific filtering goals without the risk of feedback causing undesirable oscillations or instability. One of the standout features of FIR filters is their ability to achieve linear phase characteristics. By designing the coefficient sequence to be symmetric, the phase response of the filter can be made linear, ensuring that all frequency components of the signal experience the same delay. This is particularly important in applications where phase distortion must be minimized, such as in high-fidelity audio processing or data communications. Linear phase FIR filters ensure that signals are processed without introducing phase shifts that could distort the signal, making them highly suitable for phase-sensitive applications.

In addition to their stability and phase characteristics, FIR filters also offer significant flexibility in terms of frequency response design. Unlike IIR filters, which are more prone to ripple effects in the frequency response due to their feedback structure, FIR filters can be designed to provide a smooth, well-behaved frequency response with minimal ripples. This makes FIR filters ideal for applications requiring precise control over the filtering process, such as in noise reduction, signal smoothing, and other signal conditioning tasks.

## 2. LITERATURE SURVEY

**Yadav Ranjeeta, et al. [1]** implemented an FIR filter by using a Han-Carlson adder. To design a filter, different blocks are required, which are adders, multipliers, and delay elements. FIR filters are easy to design and are less power-consuming. Here, the design of the multiplier uses the Han-Carlson Adder, and the filter is designed using the proposed multiplier and delay element (D-flip flop). The reason for selecting the Han-Carlson adder is its performance; it is one of the fastest adders compared to other adders. On the other hand, its circuit is complex, but it has given efficient performance compared to other adders. Our focus is on the parameters of adders, multipliers, and FIR filters such as power, LUT utilization, area, and delay to make an efficient FIR filter using Vedic mathematics. The proposed FIR filter has been designed and

International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

simulated with help from Verilog HDL codes and their circuit diagram, and the simulation result of the filter has been synthesized on Xilinx Tool and Questa Sim.

**Syamala Devi, et al. [2]** explained that in today's digital signal processing (DSP) applications, power optimization is one of the most

significant design goals. The digital finite duration impulse response The FIR filter is recognized as one of the most important components of DSP, and as a result, researchers have done numerous significant studies on the power refinement of the filters. FIR filters with conventional multipliers and adders are used, through which the power consumption is increased. Data-driven clock gating (DDCG) is a technology that uses multibit flip-flops (MBFFs) to share a single clock enabling signal, optimize clock latency, manage clock skew, and improve routing source utilization. To achieve low-power consumption, these two low-power design strategies are integrated. The Xilinx ISE 14.7 software is used to accomplish low-power optimization.

**Iqbal, et al. [3]** In communication system applications, the need for efficient design and implementation of the finite impulse response (FIR) filter is essential. Realization of such a filter with high bit width is a challenging task. The multiplier and accumulator blocks of the FIR filter take more delay, power, and area. In this literature the design of the FIR filter using the efficient multiplier and accumulator blocks is presented. They propose to use operations such as right shift, left shift, and carry save adder to develop the FIR filter. The multiplication and division are performed using the right shift and left shift operations. The FIR filter using the modified constant shift method (MCSM) multiplier block instead of a conventional multiplier is proposed. The modified multiplication decreases the quantity of hardware, power consumption, and surge the performance of the filter architecture. The FIR architecture is implemented using Xilinx Vivado. The parameter such as the look-up table (LUT) count, slices, flip-flops, power, and speed, are analysed with respect to existing filter architectures. Thus, the proposed novel filter architecture is implemented with high speed,less area, and less power.

**Gayathri et al. [4]** explained that in real-life applications, the signals are continuously captured, monitored, processed, and analyzed. The processing and analysis of data is easier if it is in the form of digital. The Digital Signal Processing (DSP) finds importance mainly in biomedical devices or wearable devices. In a DSP system, the FIR filter design acts as a basic building block. In wearable applications where complex computation is involved, to acquire high accuracy, the filter with higher order is used. The multipliers are the heart of any filter design, accommodate major chip area, and require extra time for computation. The designers mainly concentrate on the optimization of the multiplier over the existing one. An attempt is made in this paper to design an FIR filter based on the Distributed Arithmetic (DA) algorithm, which mainly depends on the precomputed values stored in the Look-Up Table (LUT). It is a multiplier-less design architecture. It is observed that the distributed arithmetic-based architecture is efficient for real signal computation. The advantages of the DA technique over the traditional MAC-based design. Both designs are coded in Verilog and verified for functionality, and a comparison is made. The proposed design has given an area, power, and timing of 64%. 62% and 61%, respectively.

**Kumar et al. [5]** designed architecture for a FIR filter proposed in this study to efficiently reduce noise in electrooculography (EOG) signals. Electromagnetic interference (EMI) and muscular activity are two common sources of noise that regularly alter EOG signals, which are utilized to identify eye movements. Real-time applications were challenging for conventional FIR filter architectures because of their restrictions on power consumption and delay. By utilizing a state-of- the-art technique for coefficient quantization that reduces the number of multipliers required in the filter, the recommended architecture gets around these shortcomings. The systolic architecture allows for efficient parallel processing of the input signal samples, resulting in high throughput and low latency. The implementation is verified using EOG signal datasets, and the denoising performance is evaluated in terms of signal-to-noise ratio (SNR) and mean square error (MSE).

**Shanthi et al. [6]** proposed that multiplication and division operations have been extensively used as basic elements when designing a system for advanced applications. In today's digital era, speed and area are the main constraints while implementing the digital systems. Many processors use the carry select adder (CSA), one of the faster adders. To improve the efficiency of the adder used in various applications, different architectures were adopted. It is well known that processors in the semiconductor industry perform millions of work functions per second. Performance speed must therefore be taken into account as one of the major requirements while developing a multiplier. In this literature, they offer a method for designing FIR filters that makes use of carry-select adders and compressor-based multipliers. The performance of the proposed FIR filter outperformed the power and delay compared with existing FIR filters.

**Rao et al. [7]** explained that the Fast FIR algorithm (FFA) produces a reduced complexity parallel FIR filtering structure. The FFA can reduce the number of multiplications significantly for large values of filter length N. In this paper we have proposed a new approach to design 2-parallel and 3-parallel (i.e., M = 2 and 3) even-length FIR filters with poly-phase coefficient symmetry based on FFA. Replacement of multipliers by adders is beneficial in VLSI applications as adders have less weight than multipliers. Additionally, although the overhead caused by the increment of adders in the pre- processed and post-processed blocks remains constant, the number of multipliers reduces with an increase in filter length. For a 3-parallel 88-tap filter, the proposed structure saves more than 14 multipliers, whereas for a 3-parallel 578-tap filter, the proposed parallel structure saves more than 96 multipliers. Similarly, for a 2-parallel 88-tap filter, the proposed structure saves 22 multipliers, and for 578-tap it saves 144 multipliers over reported FFAs. Overall, the suggested parallel FIR structures can result in significant hardware savings over the reported FFAs, especially for longer filter lengths.

**Baker et al. [8]** introduced a finite impulse response (FIR) filter structure based on common operation sharing. Stochastic computing (SC) uses streams of pseudo-random bits to perform low-cost and error-tolerant numerical processing for applications like neural networks and digital filtering. A key operation in these domains is thesummation of many hundreds of bit-streams, but existing SC adders are inflexible and

International journal of imaging science and engineering
ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

unpredictable. Basic mux adders have low area but poor accuracy, while other adders like accumulative parallel counters (APCs) have good accuracy but high area. This work introduced parallel sampling adders (PSAs), a novel weighted adder family that offers a favorable area-accuracy trade-off and provides great flexibility to large-scale SC adder design. Our experiments show that PSAs can sometimes achieve the same high accuracy as APCs but at half the area cost. They show that mux-based adders are sometimes more accurate than APCs, which contradicts most prior studies. An explanation for these is given, and a decorrelation scheme is proposed to improve APC accuracy by 4x for a digital filtering application.

**Balaji, et.al [9]** provided design and implementation of a 4-tap, 8-tap, 16-tap, 32-tap, and 64-tap RNS (Residue Number System) based on efficient and excessive-overall performance FIR filter. RNS mathematics is a prized tool for theoretical investigation of the speed limitations of rapid mathematics. Some suggested solutions also include a few addition operations; however, using conventional adders will slow down operation and add to the amount of logic gates. So, to address the aforementioned concerns, Kogge-Stone Adder and Brent Kung Adder are being used to reduce delay and area and enhance performance as a whole. First, the multiplier is created using the RNS methodology. In which the Vedic multiplier's power dissipation is also minimized while the latency is shortened from 70% to 90%. In order to assess the findings, we are also using a simple adder and a simple multiplier. Using the Quartus 9.0 Simulation Tool, the combination of those methods results in a completed new structure with an excessive high speed and a small implementation area for the FIR filter

**Biswas, et.al [10]** suggested that FIR filters are known for their stability and the reason being widely used over Infinite Impulse Response (IIR) filters. Out of many FIR filters, parallel FIR filters are chosen the best over other filters in digital signal processing. This paper compares the design of a basic 3-parallel FIR filter as well as two versions of Area-Efficient 3-FIR filters. Booth's algorithm and Carry-look-ahead adder are used as the basic components of this filter. The Area-Efficient filters are later compared after design them. The simulation has been carried out using Xilinx ISE design suite 14.7. TheFIR filters are later implemented on Spartan XC6SLX16-2FTG256 asthe FPGA hardware.

**Bhagavatula, et.al [11]** demonstrated the usage of forest optimization technique for the determination of optimal parameters for finite impulse response (FIR) filter. Inputs are selected as design specifications; an attempt to apply the forest optimization-based algorithm for the complex nonlinear, constrained optimization task of design the filter has been made. Optimization of filter parameters is done by exploiting the natural evolution of trees in a forest. Local seeding is employed for local searched concept ensured that only global solution is reached without falling into local optimum. Unlike the conventional windowed techniques generally employed, proposed technique realized the ideal frequency specified targets related to passband filter and stop band. Simulation results are presented.

**Farag, et.al [12]** proposed interpretations enable the employment of CLs to develop finite impulse response (FIR) filters, matched filters (MFs), short-time Fourier transform (STFT), discrete-time Fourier transform (DTFT), and continuous wavelet transform (CWT) algorithms. The main idea is to pre-assign the CL kernel weights to implement a specific convolution- or correlation-based DSP algorithm. Such an approach enables building self-contained DNN models in which CLs are utilized for various preprocessing and feature extractions tasks, enhancing the model portability, and cutting down the preprocessing computational cost. The proposed DSP interpretations provide an effective means to analyse and explain the operation of automatically trained CLs in the time and frequency domains by reversing the design procedures. The presented interpretations are mathematically established and experimentally validated with a comprehensive machinery fault diagnosis application example illustrating the potential of the proposed methodology.

## 3. PROPOSED METHODOLOGY

The implementation of FIR filters using inner product units and parallel accumulations has significantly improved signal processing efficiency. Traditionally, FIR filters involved complex sequential computations, leading to slower processing speeds. However, inner product units, which perform vector multiplications and summations, enable faster convolution operations, enhancing real-time filtering capabilities. Parallel accumulations further boost efficiency by distributing computations across multiple paths, allowing simultaneous processing and reducing delays. Additionally, coefficient storage units (CSUs) play a crucial role in managing filter coefficients efficiently. Instead of static memory storage, modern CSUs use parallel register banks, which partition and store coefficients for optimized access, reducing memory constraints. Inner product units, essential for FIR filters, perform dot product calculations using multipliers and logic operations, improving speed and power efficiency. Their ability to parallelize computations through bitwise AND operations makes real-time signal processing possible. This innovative approach enhances FIR filter performance, making them suitable for high-speed, resource-efficient applications.

Implementing FIR filters through the integration of inner product units and parallel accumulation mechanisms represents a sophisticated approach that offers notable advantages in computational efficiency and real-time signal processing. The core concept involves leveraging inner products, where corresponding elements of two vectors are multiplied and summed in a highly parallelizable fashion. This parallel processing capability, combined with parallel accumulations, allows for the simultaneous computation of multiple data points, significantly reducing latency in real-time applications. The simplicity of the mathematical operations involved facilitates straightforward implementation in both software and hardware environments. Moreover, this approach aligns well with modern processor architectures, enabling vectorization and efficient use of available hardware resources. FIR filters designed with inner product units and parallel accumulations are particularly well-suited for applications requiring low latency, such as audio processing, communication systems, and control systems. This method not only enhances computational efficiency but also proves adaptable to resource- constrained environments, making it an ideal choice for embedded systems, including IOT devices and wearable technology. In essence, the integration of inner product units and parallel accumulation mechanisms stands as a powerful strategy, delivering both speed and efficiency in the processing of finite impulse response filters across diverse applications.

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

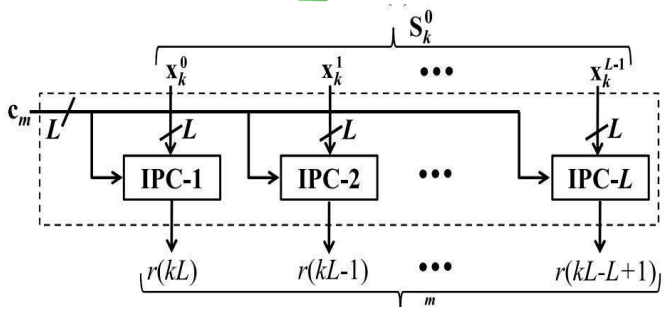


**Figure 1: Inner Product Units**

- The implementation of FIR filters using inner product units and parallel accumulations has significantly improved signal processing efficiency. Inner product units perform the convolution operation by multiplying corresponding elements of two vectors and summing the products, enabling faster and moreaccurate filtering. Parallel accumulations further enhance performance by dividing the filtering
- process into multiple parallel paths, allowing independent calculations and efficient workload distribution across multiple processing units. Additionally, the coefficient storage unit (CSU) plays a crucial role in managing filter coefficients. Instead of relying on traditional ROM storage, an advanced CSU design utilizes parallel register banks to efficiently store and access coefficients. This approach reduces memory constraints and improves real-time processing by allowing simultaneous retrieval and utilization of coefficients, ultimately enhancing the speed and accuracy of FIR filters in applications such as telecommunications, image processing, and biomedical engineering.
- The integration of inner product units and parallel accumulations in FIR filter design has revolutionized digital signal processing by significantly enhancing speed and efficiency. Inner product units execute convolution operations by multiplying and summing vector elements in parallel, reducing computational latency and enabling real-time processing. Meanwhile, parallel accumulations distribute filtering tasks across multiple processing paths, ensuring optimal workload management and boosting overall performance. Additionally, the coefficient storage unit (CSU) plays a vital role in efficient coefficient management. Traditional memory-based storage systems often face limitations due to high memory requirements, but the use of parallel register banks in the CSU overcomes this challenge. By partitioning coefficients into smaller subsets and storing them in dedicated register banks, the FIR filter ensures rapid access, reduces memory bottlenecks, and minimizes processing delays. These advancements make FIR filters more effective for applications such as telecommunications, medical imaging, and audio signal enhancement, where high-speed, high-precision filtering is essential.
- The Inner Product Unit (IPU) is a computational unit or hardware component designed to perform the inner product operation using multipliers. The inner product, also known as the dot product, is a mathematical operation that takes two vectors and returns a scalar value. This operation is widely used in fields such as linear algebra, signal processing, and machine learning. In hardware design, an IPU is optimized to efficiently execute this operation, particularly in applications like deep learning, where matrix multiplications are essential. The IPU consists of multiple inner product cells, each responsible for computing a partial inner product of specific vector elements, enabling parallel processing to enhance performance.
- The register unit in an FIR filter is crucial for handling input data efficiently. Initially, the input data is stored in D flip-flops, ensuring stability before processing. The register unit then applies a data shifting operation, where bits are delayed based on the filter coefficients. Each delay stage produces a different version of the input data, helping in generating the required output. The process continues for all bits, maintaining synchronization between input and delayed signals. The unit operates in parallel, optimizing computational efficiency while reducing latency, making it suitable for real-time signal processing applications.
- Pipeline adder units enhance computational efficiency by breaking down the addition process into multiple stages. Each stage processes a part of the addition operation and passes intermediate results to the next stage, significantly improving throughput. The implementation involves D flip-flops for synchronization, ensuring stable transitions between pipeline stages. These units comprise input registers, partial sum generation, carry computation, carry propagation, and output storage. By parallelizing the computation, pipeline adders reduce delays, making them ideal for high-speed applications like modern processors and digital signal processing.
- The FIR filter operates through coordinated interaction between different units. First, filter coefficients are preloaded into storage registers, ensuring consistency throughout processing. The input data is then stored in register units, accumulating a block before filtering begins. The inner product unit calculates the inner product between input samples and filter coefficients, generating partial products. These are then processed by a pipelined adder unit, which accumulates the results efficiently. Finally, the filtered output is stored or transmitted for further processing. Control logic ensures proper sequencing, while parallelism and pipelining optimize performance, making the FIR filter suitable for real-timeapplications.

**Applications:**

**Digital Signal Processing (DSP)-Used** in noise removal, speech recognition, and communication systems like 5G and Wi-Fi.

**Machine learning & AI speed up** matrix multiplications in neural networks, improving deep learning performance.

**Image & Audio Processing Enhances** image quality, removes noise,and improves sound clarity in devices like cameras and hearing aids.

**Advantages:**

International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

**High Computational Efficiency and Speed:**
Inner Product Units (IPUs) and pipelined FIR filters significantly enhance computational efficiency by enabling parallel processing. By dividing operations into multiple smaller tasks and executing them concurrently, these units achieve faster processing speeds compared to traditional methods. This makes them highly suitable for applications requiring real-time computations, such as signal processing, machine learning, and high-performance computing. The ability to execute multiple multiplications and accumulations simultaneously allows systems to handle large datasets efficiently without major latency issues.

**Reduced Power Consumption and Optimized Hardware:**
Power efficiency is a critical aspect of modern digital systems, and IPUs contribute to energy savings by using optimized logic circuits for inner product calculations. Instead of relying on multiple general- purpose multipliers, which consume high power, dedicated IPUs perform the same tasks more efficiently with reduced power consumption. This advantage is particularly beneficial for portable devices, embedded systems, and battery-powered electronics, where extending battery life is essential. Additionally, FIR filters designed with efficient hardware architectures minimize redundant computations, further conserving power.

**Improved Accuracy, Stability, and No Phase Distortion:** FIR filters offer a distinct advantage over infinite impulse response (IIR) filters due to their inherent stability and lack of phase distortion. Since FIR filters rely on a finite number of past input samples, they do not introduce feedback loops, preventing issues like signal instability. Moreover, their linear-phase characteristics ensure that all frequency components of a signal are delayed equally, preserving the waveform's integrity. This property is crucial in applications like audio processing, radar systems, and biomedical signal analysis, where accuracy and signal fidelity are paramount.

Inner Product Units (IPUs) and FIR filters provide significant advantages in computational efficiency, power optimization, and signal accuracy. By enabling parallel processing, IPUs enhance speed and reduce latency, making them ideal for real-time applications like signal processing and machine learning. Additionally, their optimized hardware design minimizes power consumption, which is particularly beneficial for battery-operated and embedded systems. FIR filters further contribute to improved signal integrity by ensuring stability and eliminating phase distortion, making them widely used in applicationssuch as audio processing, radar systems, and biomedical signal analysis.

## 4. EXPERIMENTAL ANALYSIS

The experimental evaluation of FIR filter architectures was conducted using Xilinx Vivado. The study compared the performance of the existing and proposed systems based on key metrics such as area, power consumption, and delay.

The existing FIR filter design utilized ripple carry adders and array multipliers. While functional, this architecture exhibited limitations, including high power consumption, increased hardware complexity, and significant delay. Simulation results highlighted inefficiencies, making the design unsuitable for resource-constrained or high-speed applications.

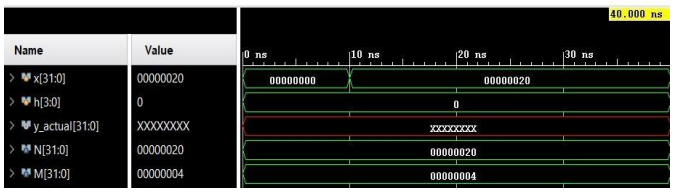


**Figure 1: Existing Simulation Result**

| Resource | Estimation | Available | Utilization... |
|---|---|---|---|
| LUT | 136 | 134600 | 0.10 |
| IO | 67 | 500 | 13.40 |

**Figure 2: Existing Area**

The proposed FIR filter incorporated pipelined adders, coefficient storage units, and parallel processing techniques to enhance performance. These modifications achieved:

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

| Resource | Estimation | Available | Utilizatio... |
|---|---|---|---|
| LUT | 81 | 134600 | 0.06 |
| LUTRAM | 32 | 46200 | 0.07 |
| FF | 99 | 269200 | 0.04 |
| IO | 67 | 500 | 13.40 |
| BUFG | 1 | 32 | 3.13 |

**Figure 3: Area Reduction-LUT usage decreased by 30%.**

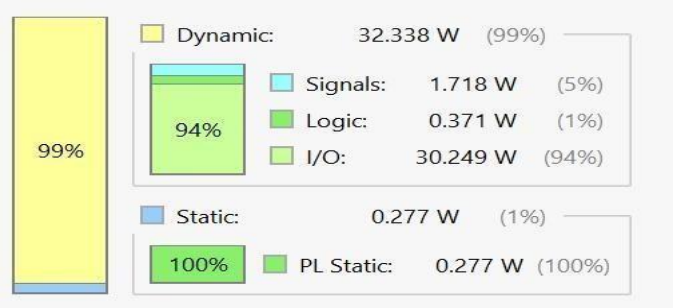


**Figure 4: Lower Power Consumption - Total power consumptionreduced by 20%.**

## 5. CONCLUSION

In conclusion, the register unit within an FIR filter serves as the cornerstone of efficient signal processing, orchestrating a series of meticulously designed steps to transform input data into a filtered output. From the initial storage of input data in D flip-flops to the subsequent data shifting operations and generation of coefficients, each step in the register unit's operation is geared towards achieving precise and tailored filtering. The unit's ability to simultaneously process multiple streams of data in parallel, each subjected to different levels of delay, underscores its role as an adept parallel processing entity within the FIR filter architecture. This parallelism not only enhances computational efficiency but also contributes to minimizingprocessing latency, a critical aspect in real-time signal processing applications.

Furthermore, the register unit's commitment to uniform data delay across all input bits ensures the preservation of synchronization and coherence in the output data. The integration of D flip-flops, logical operations, and coefficient generation showcases a harmonious blend of digital circuit fundamentals and advanced filtering techniques. In essence, the register unit's meticulous orchestration of these steps facilitates the creation of filtered output streams that faithfully represent the input data while accommodating the defined filter characteristics. Thus, the FIR filter's register unit stands as a testament to the intersection of digital circuitry and signal processing prowess, playing a pivotal role in the realization of precise and responsive filtering in diverse applications ranging from telecommunications to audio processing.

## REFERENCES

[1]. Yadav Ranjeeta, Surekha Gangas, Krishna Kumar Verma, Divyanshu Joshi, Harsit Yadav, and Anmol Dev. "IMPLEMENTATION OF EFFICIENT FIR FILTER."
International Development Planning Review 22, no. 2(2023): 9-20.

[2]. Syamala Devi, P., D. Vishnupriya, G. Shirisha, Venkata Tharun Reddy Gandham, and Siva Ram Mallela. "Design of High Efficiency FIR Filters by Using Booth Multiplierand Data-Driven Clock Gating and Multibit Flip-Flops."
In International Conference on Communications and Cyber Physical Engineering 2018, pp. 319-326. Singapore: Springer Nature Singapore, 2023.

[3]. Iqbal, JL Mazher, G. Narayan, T. Manikandan, M. Meena, and Jose Anand. "Low power and low area multiplier and accumulator block for efficient implementation of FIR filter." In Low Power Designs in Nanodevices and Circuits for Emerging Applications, pp. 267-282. CRC Press.

[4]. Gayathri, S., S. Esha, Challa Bhavya, and Yasha Jyothi M. Shirur. "Design and Implementation of Arithmetic-Based FIR Filters for DSP Application." In 2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE), pp. 782-787. IEEE, 2023.

[5]. Kumar, A., Ramesh, Aruru Sai Kumar, K. Hemanth Lakshmi Phani Prasad, B. Sriraj, and P. Raja Rajasri. "High performance FIR Architecture for EOG Signal Noise Suppression." In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-6. IEEE, 2023.

[6]. Shanthi, G., Aruru Sai Kumar, Md Masood Hasan, H. Tanuja, and Ch Yashwanth. "An Efficient and High-Speed FIR Filter using BEC with MUX Technique." In 2023 3rd International Conference on Advances in Computing, Communication, Embedded and Secure Systems

(ACCESS), pp. 256-262. IEEE, 2023.

[7]. Rao, K. Anjali, and Niteesh Purohit. "Hardware Efficient 2- Parallel and 3-Parallel Even Length FIR Filters Using FFA." In 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), pp. 1-5. IEEE, 2023.

[8]. Baker, Timothy, and John P. Hayes. "Design of Large-Scale Stochastic Computing Adders and their Anomalous Behavior." In 2023 Design, Automation & Test in EuropeConference & Exhibition (DATE), pp. 1-6. IEEE, 2023.

[9]. Balaji, M., N. Padmaja, P. Gitanjali, Saif Ali Shaik, and Siva Kumar. "Design of FIR filter with Fast Adders and Fast Multipliers using RNS Algorithm." In 2023 4th International Conference for Emerging Technology (INCET), pp. 1-6. IEEE, 2023.

[10]. Biswas, Neelesh, Supriya Dhabal, and Palaniandavar Venkateswaran. "Analysis of Area Efficient Parallel FIR Filters using FPGA." In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-5. IEEE, 2023.

[11]. Bhagavatula, Venkata Vaibhav, and S. V. N. L. Lalitha. "Optimization of FIR filter parameters using forest optimization algorithm." In AIP Conference Proceedings,vol. 2512, no. 1. AIP Publishing, 2024.

[12]. Farag, Mohammed M. "Design and Analysis of Convolutional Neural Layers: A Signal Processing Perspective." IEEE Access 11 (2023): 27641-27661.