iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VI **Month of publication:** June 2022

DOI: https://doi.org/10.22214/ijraset.2022.43516

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

A Review of Techniques for Optimization and Implementation of Digital Filters on FPGA

Ayushi Pandey[1], Er. Pratibha[2]
[1, 2]*Department of Electronics & Communication Engineering, Shri Ramswaroop Memorial College of Engineering and Management, Lucknow, India*

Abstract: ***The use of digital filters extends to various fields. Obtaining a response that is close to the desired response is a major goal of designing digital filters. Designing of filters involves providing the specification, determining coefficients and, then the realization of filter on hardware. This paper provides an insight into the algorithms used for determining optimized filter coefficients, the realization of the filter on hardware and, some methods to reduce the hardware usage by the filter***
Index Terms: ***FPGA, FIR, IIR, MSE.***

I. INTRODUCTION

Digital Signal Processing has become a vital part of most electronic systems, and the reason for this is that digital processing methods can perform far more complex operations than analog ones. For example, digital filters of higher orders can be simply designed without occupying any extra 'space', which is a major limiting factor in analog systems. Another thing is that digital filters do not depend on environmental parameters so they can work flawlessly without any faults for a long time.

Digital filters work on sampled analog data and can be categorized into two types Infinite Impulse Response (IIR) and Finite Impulse Response (FIR). Now the question arises how these filters are implemented? These filters are constructed using various blocks like adders, multipliers and, delay elements that can be implemented on a digital signal processor or an FPGA. These implementations become more flexible with FPGA where the user can very simply program the logic to be implemented. However, there is a limitation of the number of logic blocks in an FPGA so filters are needed to be designed to occupy minimum LEs. An inefficient filter can be very slow in its operation which is a critical problem in real-time or high-speed applications.

The designing of Digital Filters involves the determination of appropriate filter coefficients that give the desired response and the realization of the filter on the hardware such as FPGA in our case. This paper gives a review of the various optimization algorithm to reduce the error difference between the output and the ideal response of the filter and the determination of optimized coefficients of the filters. Then the paper provides an insight into the methods proposed to reduce the hardware usage by the filters and to increase the processing speeds of the filters.

Part II and III of this paper describe the FIR and IIR filter

II. FIR FILTERS

These are non-recursive filters have no feedback and depend only on past and present input values. The FIR filters have a linear response and are stable. However, these filters require more hardware for realization as compared to the IIR filter for the same response. In other words, to obtain a similar kind of response FIR filter require more order than the IIR filters.

The output of the FIR filter is given by:

$$y[n] = \sum_{k=0}^{N-1} h_k . x[n-k] \quad (1)$$

where N denotes the order of the filter.

The condition for linear response of FIR filter is

$$\text{even symmetric condition: } h(n) = h(N-n-1) \quad (2)$$

or

$$\text{odd symmetric condition: } h(n) = -h(N-n-1) \quad (3)$$

and

$$\tau = \frac{N-1}{2} \quad (4)$$

Where τ is the constant phase delay that is expressed in terms of the number of samples.

When the above equations are satisfied then the FIR filter will have constant phase and group delay and the filter has a linear response.

There are 3 main methods of designing FIR filters

A. Frequency Sampling Method

In this method, the Impulse response of the filter is obtained by sampling the desired frequency response. The samples are taken at the frequency

$\omega = \frac{2k}{N} \quad where\ k = 0,1,\dots\dots,N-1$ (5)

Samples of the desired frequency response are given by:

$H(k) = H_d\left(e^{j2\pi k/N}\right) \quad where\ k = 0,1,\dots.N-1$ (6)

These set of samples are considered as DFT samples and the inverse discrete Fourier transform of H(k) is carried out to determine the impulse response (Filter coefficients) of the FIR filter

$$h(n) = \frac{1}{N}\sum_{k=0}^{N-1} H(k)e^{j2\pi nk/N} \quad where\ n = 0,1,\dots\dots N-1 \quad (7)$$

B. Windowing Technique

The Desired impulse response can be obtained by taking the Inverse Fourier transform of the desired filter response but this impulse response is infinite in duration thus it is not suitable for realization. In the window technique, the impulse response is multiplied by the window function

$h(n) = h_d(n)w(n)$ (8)

In the frequency domain, it is given by the convolution of the desired frequency response and the window function

$H(e^{j\omega}) = H_d(e^{j\omega}) * W(e^{j\omega})$ (9)

Some of the commonly used windows and their window function are mentioned as follows:

RECTANGULAR WINDOW

$$W_R(n) = \begin{cases} 1, for\ |n| \le \frac{N-1}{2} \\ 0,\ otherwise \end{cases} \quad (10)$$

HAMMING WINDOW

$$W_H(n) = \begin{cases} 0.54 - 0.46\cos\frac{2\pi n}{N-1}, 0 \le n < N-1 \\ 0,\ otherwise \end{cases} \quad (11)$$

HANNING WINDOW

$$W_{Hann}(n) = \begin{cases} 0.5 - 0.5\cos\frac{2\pi n}{N-1}, 0 \le n < N-1 \\ 0,\ otherwise \end{cases} \quad (12)$$

BLACKMAN WINDOW

$$W_B(n) = \begin{cases} 0.42 - 0.5\cos\frac{2\pi n}{N-1} + 0.08\cos\frac{4\pi n}{N-1}, 0 \le n < N-1 \\ 0,\ otherwise \end{cases} \quad (13)$$

C. Optimal Linear Phase FIR Filter

With the Frequency Sampling and windowing technique, precise control of the critical frequency is a problem. In this, the filter coefficients are determined such that they minimize the maximum absolute value of the approximation error which is the error between the actual response and the desired frequency response. The weighted error function is given as follows

$E(\omega) = W(\omega)[H_d(\omega) - H(e^{j\omega})]$ (14)

III. IIR FILTERS

The IIR filters are recursive containing feedback paths as they depend on past output values. These filters require lesser multiplier units in comparison to the FIR filters and have a response similar to their analog filter models. The IIR filters have a better magnitude response and require lesser coefficients as compared to the FIR filters providing the same response. IIR filters realize both the poles as well as zeroes of its transfer function. The output is given by the difference equation as follows:

$y(n) = \sum_{k=0}^{M} a_k.x[n-k] - \sum_{i=0}^{N} b_i.y[n-i]$ (15)

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

IV. CONCEPT OF ADAPTIVE FILTERS

Adaptive filters are the self-adjusting filters that give the optimized coefficients of the filter which minimize the mean square error (MSE) between the desired response and the obtained response. With the change in desired response, the filter provides the new optimized coefficients of the filter.

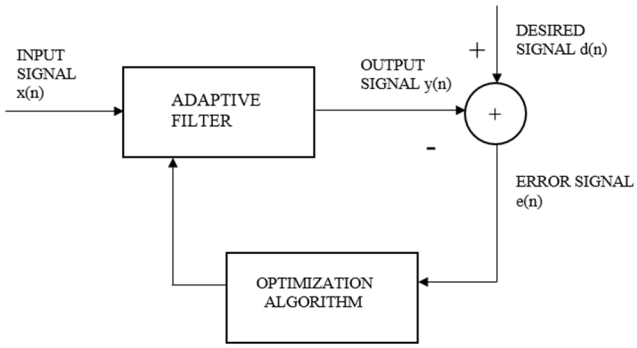


Fig. 1 Block diagram of Adaptive Filter

The objective function Mean Square Error (MSE) is [9]:

$$J_{a_i,b_i} = \frac{1}{S}\sum_{i=1}^{S}(d(n) - y(n))^2 \qquad (16)$$

Where S denotes the number of samples that are used for coefficient computation.

V. OPTIMIZATION OF DIGITAL FILTERS

Designing of filter involves determining the filter coefficients that will provide the desired performance of the filter and satisfy the features such as stopband attenuation, pass band and stop band ripples, Cut off frequency, the width of the passband, and the stopband. There are many conventional techniques of determining the filter coefficients in IIR and FIR filters such as for FIR Filters some of the conventional techniques namely, Frequency Sampling method, Windowing techniques such as Hamming window, Hanning Window, Barlett Window, etc. and for the designing of IIR filters, some of the conventional techniques are transformation techniques like bilinear transformation, etc.

For the designing of the filters, it is required to have Preliminary knowledge of the statistical features of the input data. While using the digital filters there are various applications where there is unsatisfactory knowledge of the input data or the statistical variations of the input data. Therefore, Adaptive filters are required. The filter parameters are adjusted to reduce the mean squared error (MSE) between the output of the filter and the desired output. Some applications require large order of FIR filter. In such cases, the use of an adaptive IIR filter is beneficial. An adaptive IIR filter has much better performance than an adaptive FIR filter but the advantages come with several difficulties like the possibility of instability of the filter during the adaptation process, slow convergence, and multi-model error surface. Hence, to reduce the error an optimization algorithm is required.

An Optimization Algorithm is used to find the best possible solution to a problem from the different accessible solutions. It is a process that is executed repeatedly till the best solution is found. [11] There are two optimization approaches. Deterministic algorithm and stochastic algorithm. The Deterministic approach requires huge computational work and there are chances of their failure when the size of the problem increases. Therefore, a metaheuristic stochastic algorithm is a better alternative to the deterministic approach. The Metaheuristic algorithms are nature-inspired phenomena and these are problem independent.

Metaheuristic techniques are very popular. Evolutionary Algorithm (EA) and Swarm Intelligence (SI) are the two popular classifications of meta-heuristic. Genetic Algorithm (GA), The Evolutionary Strategy (ES), Differential Evolution (DE) and, Biogeography-Based Optimizer (BBO) are examples of evolutionary algorithms.

Swarm Intelligence (SI) is inspired by the mutual social activities of living species such as fishes, insects, etc. These algorithms work on the mutual actions of the species rather than competition amongst themselves. Whale Optimization Algorithm (WOA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Cuckoo Search (CS), Artificial Bee Colony (ABC), Dragonfly Algorithm (DA) are examples of Swarm Intelligence.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

VI. REVIEW OF SOME PREVIOUS WORKS ON OPTIMIZATION TECHNIQUES FOR DIGITAL FILTERS

Suman, Richa, Ashwni, and Manjeet [1] provided a Grasshopper Optimization Algorithm (GOA) for designing the FIR filter. The approach focuses on the minimization of the absolute error difference fitness function to obtain the optimal filter coefficients. This method is implemented in Low pass, High pass, Bandpass, and Bandstop FIR filter and the analysis is performed. It concluded with GOA being the best choice for designing a 20^{th} order FIR filter with lesser ripples in passband as well as the stopband and higher stopband attenuation in comparison to other algorithms.

Peng, Hu, and Yang [2] proposed an Improved Grasshopper Optimization Algorithm (IGOA). The paper also compared the proposed IGOA with GOA and various other algorithms. The waveform showed that IGOA outperformed GOA in most functions and the convergence speed of IGOA was greater than that of GOA and PSO.

Singh, Ashok, Kumar M., Garima, and Rawat T.K. [9] presented Dragonfly Algorithm for the optimal design of IIR filters. Results of the DA are compared with Cat Swarm Optimization (CSO), particle swarm optimization (PSO), and bat algorithm (BA) and prove to be more efficient than the other three optimization algorithms.

Irfan, Arindam, Raina, Supriya, and Palaniandavar [10] proposed Salp Swarm Algorithm for IIR design. The paper gives a comparison of the estimated value of coefficients and the MSE of the Salp Swarm Algorithm with the Whale Optimization Algorithm and Dragonfly Algorithm which indicates outperformance of the Scalp Swarm Algorithm.

Dash, Dam, and Swain [12] used hybrid differential evolution particle swarm Optimization (HDEPSO) for the design and implementation of sharp edge FIR filters. It shows that the proposed HDEPSO performed better than PSO and Differential Evolution (DE) Algorithms under certain circumstances.

Shubhendu, Rutuparna, and Ajith [13] provided a design for optimal low pass filter by Levy swallow swarm algorithm. It compares the proposed algorithm with e firefly algorithm (FA), the sine cosine algorithm (SCA), real coded genetic algorithm (GA), conventional particle swarm optimization (PSO), cuckoo search (CS) and, SS algorithm and shows that Levy Swallow algorithm outperforms GA, SS, PSO, SCA, FA and, CS. It concludes that the proposed algorithm is suitable for use in designing FIR filters.

Liang and Kwan [14] proposed the Design of FIR filters with the use of Multiobjective Cuckoo Search Algorithm. The algorithm is applied for the optimization of the filter coefficients of FIR low pass and Band Pass Filters.

Deny, Sun, Zhang [15] used a new algorithm named ESA-DE. The proposed algorithm improves the basic DE algorithm and results show the proposed improved algorithm outperforms DE, jDE, and ODE algorithms.

Hang, Dong; Li, Xiaoyi [16] proposed a design of FIR filter that is based on Firefly Position Optimization and Improved Particle Swarm Optimization Algorithm. The proposed design had smaller oscillations and better convergence performance than PSO

VII. REALIZATION OF DIGITAL FILTERS

When the system function H(z) or impulse response h(n) is known, the digital filter can be synthesized and implemented in hardware. The difference equation requires adders, delay elements and, multipliers for its hardware implementation. There are several structures for FIR and IIR. The basic structures for IIR filter realization are:

1) Direct form realization: this includes direct form 1 and direct form 2 structure.
2) Cascade Realization
3) Parallel Realization
4) Transposed Structure
5) State Space Structure
6) Ladder Structure

For FIR filter realization there are two widely used structures, they are:
a) Direct structure
b) Cascade structure.

To realize linear phase FIR filter, we consider the condition given in (2).
Therefore, for the realization of the 4^{th} order linear phase FIR filter h (0) = h (3) and h (1) = h (2).

The difference equation will become
$y(n)=h(0)[\,x(n)+\,x(n-3)]+h(1)\,[x(n-1)+x(n-2)]$

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

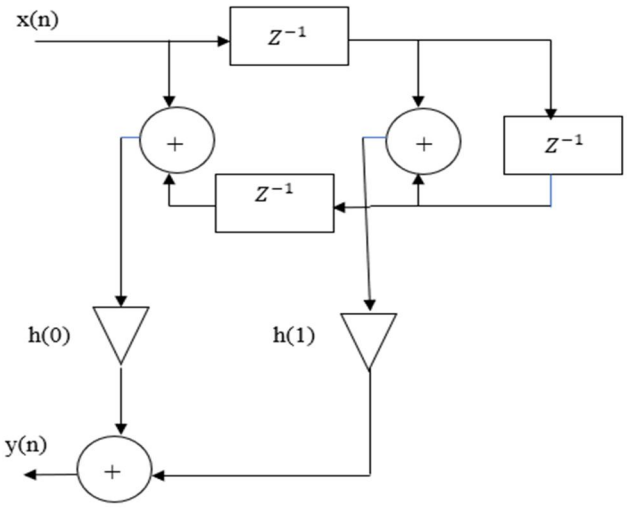


Fig. 2 Realization of 4th order linear phase FIR filter (Direct form)

VIII. A REVIEW OF HARDWARE IMPLEMENTATION TECHNIQUES FOR DIGITAL FILTERS

The implementation of FIR filters requires three components: Delay block, Multiplier, and Adder. The FIR filters have a drawback that it requires more hardware than IIR filter performing the same response. The FIR filter requires more multipliers and the major hardware is occupied by a multiplier in a filter implementation. For the implementation of digital filters on FPGA, there has been major experimentation done to achieve a greater speed of operation and low hardware and resource usage. The models used for implementation of FIR Filter on FPGA were Multiplier-based design and Distributed arithmetic-based design. The multiplier-based design uses multiply and accumulate blocks (MAC). Thus, it uses N MAC blocks for the implementation of the N tap FIR filter, which is costly to be implemented in an FPGA due to the complexity of logic.

The Distributed Arithmetic approach (DA) is a multiplier-less architecture for the implementation of the FIR filter on FPGA. This methodology involves the storage of all possible values of the coefficient of filter in the Look-Up Table (LUT) of an FPGA. This is preferable in FPGAs containing LUTs.

Keerti, Vasujadevi, Nagakishore, and Jeevan [3] provided a description of 4 architectures of the DA approach for the implementation of the FIR filter on FPGA. It begins with the traditional DA architecture and then focuses on further optimization of the architecture by reducing the hardware used with the implementation of LUT less and 4 input LUT DA. It
can be summarized that The DA approach can be used for known coefficients of the FIR filter.

Sakthimohan and deny [4] proposed an architecture of multiplier that reduced the number of steps of multiplication and reduced hardware required and the power consumption. The architecture implements Radix-4 Booth multiplier using an improved Booth recording algorithm

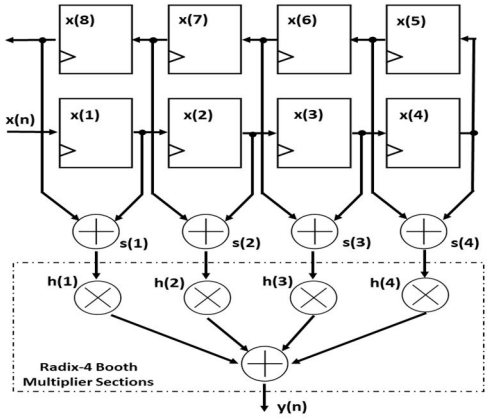


Fig. 3 Proposed Radix 4 booth multiplier-based FIR filter

Shahnam, Anup, and Ryan [5] proposed a new methodology for implementing high-speed FIR filters. It implemented the FIR filter using add and shift method. The filter is divided into a multiplier block and a delay block and the optimizations are performed in the multiplier block where the constant coefficient multiplication is decomposed into registered additions and hardwired shifts. 2 input adders are used to perform the addition and are arranged in the fastest tree structure such that the performance of the filter is affected by the slowest adder. The methodology uses the subexpression elimination technique for reducing addition operations and the number of adders to reduce the area and insertion of necessary registers to synchronize values. The method is implemented and performance is compared with the DA approach and the MAC approach of FIR implementation on FPGA and resulted in better performance of add and shift technique.

Yuan and Yanzhi [6] presented a high accuracy FIR filter design. They proposed a stochastic computing method for FIR filter design. The approach presented in the paper uses high accuracy stochastic adder and multiplier which is based on the two-line stochastic computing representation. The adder is non-scaled therefore the output of the filter is non-scaled and has high accuracy compared to conventional Stochastic FIR filters..

Seshadri, R.; Ramakrishnan, S. [17] designed 1st order and 2nd order IIR Filter using Look Ahead technique and implemented in level 1 and level 2. Then 8, 16, 32, and 64 Tap MA FIR filters are implemented using Cascaded Integrator Comb (CIC) and Look-ahead schemes. The results show that the level 2 look ahead has better performance than level 1 which in turn has better performance than conventional methods. In the FIR filter, level 2 Look Ahead outperformed CIC. The paper also compares other factors such as Les utilized by each technique and Power Dissipation.

Debarshi and Himadri [7] proposed a design for high performance IIR filter and its implementation on FPGA. It provided a design for reconfigurable IIR filters for real-time applications. It described FIR-based IIR design. It also briefly described lossy integrator-based look-ahead IIR filter, two-level parallel-pipeline IIR filter and FIR-based IIR design and implemented the designs on Xilinx Vertex 5 FPGA board. The paper experimentally concludes that the proposed FIR-based IIR filter implementation technique provided maximum operating speed, lesser power consumption, and reduced area in comparison with the look-ahead and parallel pipeline technique.

IX. CONCLUSION

There has been continuous research on finding a better algorithm for designing specific types of filters to obtain the best performance of the filter and to reduce the error between the obtained response and the desired response. For the determination of optimized filter coefficients, a wide variety of algorithms have been developed. The superiority of a particular algorithm depends on the application.

REFERENCES

[1] Yadav, Suman; Yadav, Richa; Kumar, Ashwni; Kumar, Manjeet (2020),"A novel approach for optimal design of digital FIR filter using grasshopper optimization algorithm," ISA Transactions, S0019057820303645–. doi:10.1016/j.isatra.2020.08.032

[2] Qin, P., Hu, H. & Yang, Z., "The improved grasshopper optimization algorithm and its applications," Sci Rep 11, 23733 (2021). https://doi.org/10.1038/s41598-021-03049-6

[3] M. Keerthi, Vasujadevi Midasala, S Nagakishore Bhavanam, Jeevan Reddy K, 2012, "FPGA Implementation Of Distributed Arithmetic For FIR Filter," International Journal of Engineering Research & Technology (IJERT) Volume 01, Issue 09 (November 2012),

[4] M. Sakthimohan, J. Deny, "An Optimistic Design of 16-Tap FIR Filter with Radix-4 Booth Multiplier Using Improved Booth Recoding Algorithm," Microprocessors and Microsystems (2020), doi: https://doi.org/10.1016/j.micpro.2020.103453

[5] Mirzaei, Shahnam; Hosangadi, Anup; Kastner, Ryan (2006). [IEEE 2006 International Conference on Computer Design - San Jose, CA, USA (2007.10.1-2007.10.4)] 2006 International Conference on Computer Design, "FPGA Implementation of High Speed FIR Filters Using Add and Shift Method,"308–313. doi:10.1109/ICCD.2006.4380833

[6] B. Yuan and Y. Wang, "High-Accuracy FIR Filter Design Using Stochastic Computing," 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2016, pp. 128-133, doi: 10.1109/ISVLSI.2016.63.

[7] Datta, D., Dutta, H.S., "High performance IIR filter implementation on FPGA," Journal of Electrical Systems and Inf Technol 8, 2 (2021). https://doi.org/10.1186/s43067-020-00025-4

[8] V. Lesnikov, T. Naumovich and A. Chastikov, "Multiplierless IIR Filter Design Technique," 2021 10th Mediterranean Conference on Embedded Computing (MECO), 2021, pp. 1-4, doi: 10.1109/MECO52532.2021.9460199.

[9] Singh S., Ashok A., Kumar M., Garima, Rawat T.K. (2019), "Optimal Design of IIR Filter Using Dragonfly Algorithm". In: Malik H., Srivastava S., Sood Y., Ahmad A. (eds) Applications of Artificial Intelligence Techniques in Engineering. Advances in Intelligent Systems and Computing, vol 698. Springer, Singapore. https://doi.org/10.1007/978-981-13-1819-1_21

[10] I. Ghazi, A. Das, R. M. Aich, S. Dhabal and P. Venkateswaran, "Improved Design of IIR filter using Salp Swarm Algorithm," 2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical Electronics and Computer Engineering (UPCON), 2020, pp. 1-5, doi: 10.1109/UPCON50219.2020.9376571

[11] Ravi, Renjith V.; Subramaniam, Kamalraj; Roshini, T. V.; Muthusamy,

Sundar Prakash Balaji; Prasanna Venkatesan, G. K. D. (2019),"Optimization algorithms, an effective tool for the design of digital filters; a review," Journal of Ambient Intelligence and Humanized Computing, doi:10.1007/s12652-019-01431-x

[12] J. Dash, B. Dam, R. Swain, "Design and implementation of sharp edge FIR filters using hybrid differential evolution particle swarm optimization," International Journal of Electronics and Communications (2019), doi: https://doi.org/10.1016/j.aeue.2019.153019

[13] Sarangi, Shubhendu Kumar; Panda, Rutuparna; Abraham, Ajith (2020), "Design of optimal low-pass filter by a new Levy swallow swarm algorithm," Soft Computing, doi:10.1007/s00500-020-05065-6

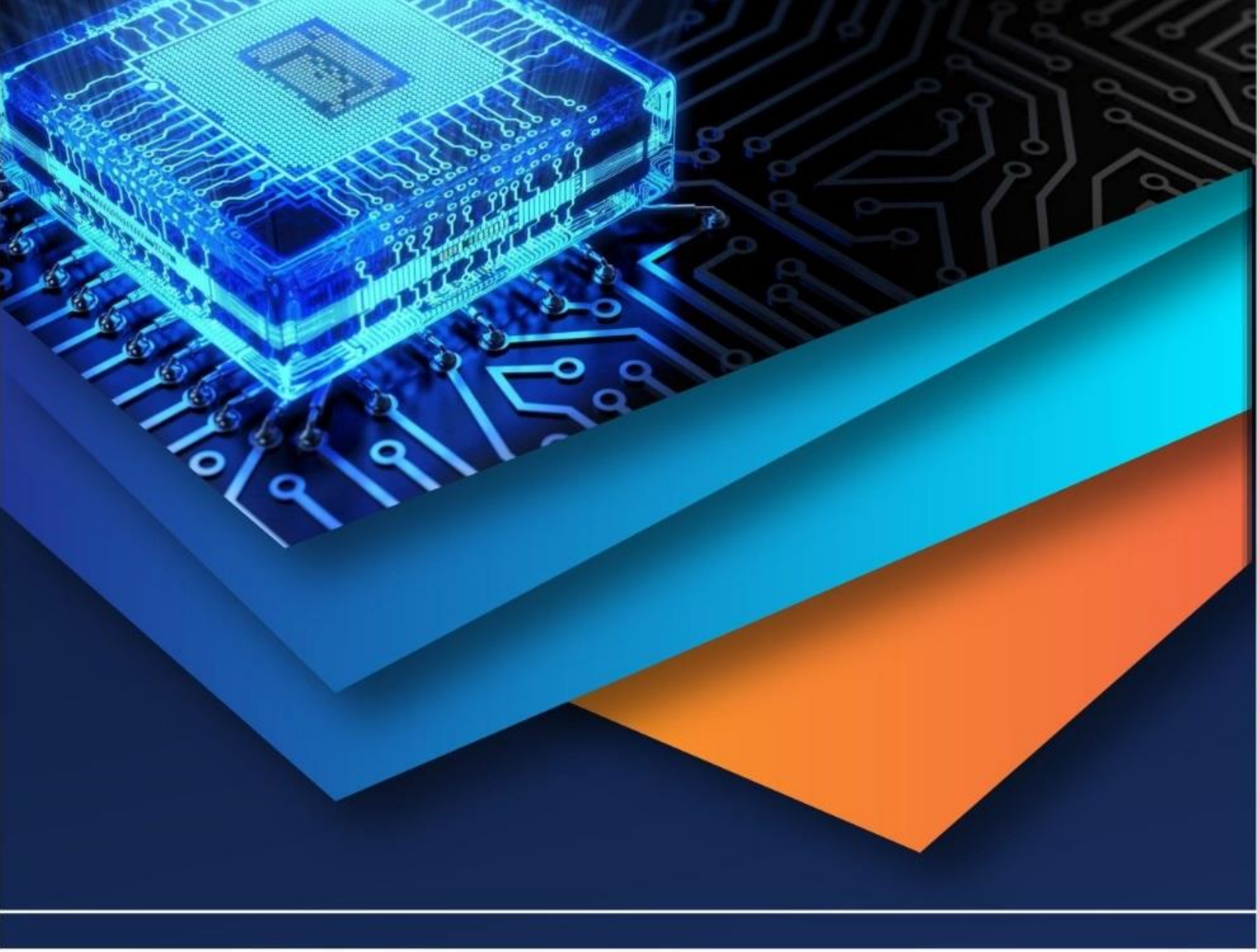
[14] J. Liang and H. K. Kwan, "FIR filter design using multiobjective Cuckoo Search Algorithm," 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), 2017, pp. 1-4, doi: 10.1109/CCECE.2017.7946828.

[15] Deng L., Sun H., Zhang L. (2019), "A New Algorithm (ESA-DE) for Designing FIR Digital Filters," In: Jia M., Guo Q., Meng W. (eds) Wireless and Satellite Systems. WiSATS 2019. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 280. Springer, Cham. https://doi.org/10.1007/978-3-030-19153-5_63

[16] Hang, Dong; Li, Xiaoyi (2018). [IEEE 2018 IEEE International Conference of Safety Produce Informatization (IICSPI) - Chongqing, China (2018.12.10-2018.12.12)] 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), "Application of Improved Particle Swarm Optimization Algorithm Based on GSO in Optimization Design of FIR Digital Filter," 84–87. doi:10.1109/IICSPI.2018.8690503

[17] Seshadri, R.; Ramakrishnan, S. (2019), "FPGA implementation of fast digital FIR and IIR filters," Concurrency and Computation: Practice and Experience, e5246–. doi:10.1002/cpe.5246

[18] S Salivahanan A Vallavaraj; Gnanapriya C., "Digital signal processing," Tata McGraw Hill New Delhi, 2000



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY